October 2012

World Wide Walk

Huan Meng
Worcester Polytechnic Institute

Jeffrey Daniel Thomas
Worcester Polytechnic Institute

Kai He
Worcester Polytechnic Institute

Sonya J. Chen
Worcester Polytechnic Institute

WORCESTER POLYTECHNIC INSTITUTE

World Wide Walk

Interactive Media and Game Development
&
Computer Science

Sonya Chen, Kai He, Huan Meng, Jeffrey Thomas

October 23, 2012

Advised by

Professor Robert W. Lindeman

Professor Mark Claypool

A Major Qualifying Project Report

Submitted to the faculty of

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

Abstract

This is a Computer Science and Interactive Media and Game Development Major Qualifying Project report focusing on the state and development of an Android application called *World Wide Walk* where users play by walking. The game merges the use of pedometers and GPS services in an attempt to curtail cheating in previous pedometer games.

This document discusses the state, development, and original design of the game *World Wide Walk*. As users play by walking, they accumulate points to gain advantage within the game world. *World Wide Walk* has unique features including a leaderboard, territory ownership, an alliance system, and a coach manager. *World Wide Walk* has two looks for User Interface (UI) and a total of twelve coaches each with eight different expressions.

Acknowledgements

Team World Wide Walk would like to thank our advisors, Professors Lindeman and Professor Claypool from Worcester Polytechnic Institute for their guidance and feedback throughout the development. We would also like to extend our thanks to Professors Takemura and Professor Kiyokawa from Osaka University for including us in Takemura Lab and taking care of our various daily life issues during our stay in Japan. Finally, we would like to thank the staff at the IGSD and the Admini Corporation, our housing agency, for facilitating our stay and making the transition to Japan easier.

Table of Contents

List of Figures

List of Tables

1 Introduction

This section explains the motivation and goals behind the making of *World Wide Walk.* It also gives an overview on the functions and user experience of *World Wide Walk*.

1.1 Motivation & Goals

The motivation behind the creation of *World Wide Walk* was to discover a solution to where many pedometer games fail. Most apps that simulate a pedometer on a phone are too inaccurate to catch cheating and in most pedometer games, rewards are usually tied to the number of steps the user walks. The inaccuracy of a pedometer on a phone often makes the game unfair since the rewards gradually become pointless as users find easier alternatives to gain points such as setting their phone or game device on top of a dryer. For example, the Pokewalker's effectiveness (see Section 2.2.4) was often curtailed by attaching the device to a box fan and letting the fan run. By creating a game with rewards and motivations for play that are not tied to the number of steps walked, a more effective exercise game could be created.

The main goal of *World Wide Walk* is to fuse pedometer and GPS functions into a walking game that is intended to be more immersive and accurate, using location check instead of step count and therefore reduce chances of cheating. *World Wide Walk* creates a parallel world of reality and has users' actions directly reflected in the game. *World Wide Walk* aimed to track not only the distance of users' walking, but also the positions so that the game can mirror users' walking on the world map. The team put more focus on the gaming aspect and designed features that encourage competition and add seriousness to the usually casual walking games. Overall, the team strived to create an immersive walking game that makes users feel that they were actually walking in the game world.

The other goal of the project is to make a game that appeals to as many people as possible. The team planned out various game features for competitive, noncompetitive, social and solo users. As for the visual styles, two sets of skins (Eastern and Western styles) were created so that users with different preferences could all find something that appeals to them.

1.2 Gameplay Overview

The main goal for the users of *World Wide Walk* is to accumulate points by walking. The points are used to determine the users' position on the leaderboard and territory ownership. Users compete with each other within pre-defined, real-world territories, and whoever has the most points earned in a territory in the previous two and a half months is awarded ownership of that territory. If users want to earn points more quickly, they may walk into new areas or complete missions to get bonus points.

For those who like to play along with friends and connect with other people, there is an alliance system which allows users to join an alliance with other users and collect points together. The team also created the coach system to bring more interaction between users and the application. The coaches are NPCs that accompany the users throughout their journey in *World Wide Walk*. There are twelve distinctive coaches, each with eight different hand-drawn expressions and unique dialogues. They provide feedback and reminders based on user performance in the game. They also offer missions which award bonus points if completed.

1.3 User Experience

World Wide Walk allows users to play anytime they are walking without interrupting their daily lives. Users would not have to devote additional time to play the game. As they go about their

day-to-day activities, such as walking to the subway station, going to school, or even a casual walk after dinner, they automatically participate in *World Wide Walk* and improve their status in the game. Users can also actively devote more time to walking, and thus collect more points.

Participating in *World Wide Walk* was designed to be an easy experience. Once a user turns on the application and starts walking outside with their Android device, they start to gather points. Users can leave their Android devices in their pockets and let the application run in the background, and they will continue to earn points. At any time, they may choose to open up the application screen to see notifications, accept missions, and/or simply receive daily greetings from the coaches. Users can also check which territories they own on the map. When they decide to stop walking, they can turn off the application and their personal data (leaderboard rank, territory ownership) will be stored in their Android device. Users' progress is synchronized and shared automatically when they have a Wi-Fi signal.

1.4 Components of the Application

World Wide Walk was built on the Android platform. The team also constructed a database on an external server located in Osaka University, Japan with MySQL to handle all the user information (points, territory ownership, and alliances). Key components like the Unique Walking Tracker, Map, GPS, Territory System and Point System track the users' data on the map, as well as compute the points they gathered through walking. The Coach Manager makes the coach pop up under different conditions with different messages and notifications; the Coach Manager also displays various hand-drawn expressions of the coaches to help users get a sense of their personalities.

1.5 Roadmap

The rest of the document is divided into five main sections: Background, Game Design, Art Development, Tech Development, Test and Conclusion. Background research on pedometers and related works can be found in Section 2: Background. Details of the design ideas and game mechanism of *World Wide Walk* are discussed in Section 3: Game Design. Detailed information about each technical component of the game is covered in Section 4: Tech Development. The process of how all the coaches are designed is explained in the Section 5: Art Development. Testing results of the game are shown in Section 6: Testing. The summary of the project is included in Section 7: Conclusion.

2 Background

In creating *World Wide Walk*, the team examined many other similar applications for inspiration and guidance. By studying the successes and learning from the failures of existing products, the team gained important knowledge to develop a better end product. Applications that were examined included Pedometer Apps and games that used steps generated by a built-in pedometer.

2.1 Pedometer Apps

A pedometer app is a computer application that utilizes GPS functionalities to count a person's walking distance. Most pedometer apps are governed by a simple pendulum system. It is a type of system that uses a pendulum that makes contact with a post when users take a step. The action of taking a step closes the circuit and registers the step in the pedometer or time-proven movements. The pedometer takes the simple pendulum system and incorporates an electronic filter circuit that prevents most non-step movements from being counted as steps.

The main example studied was the free pedometer app issued in June, 2012 by Arawella Corporation for iOS. Below are two figures (Figure 1 and Figure 2) showing the interface of this app.

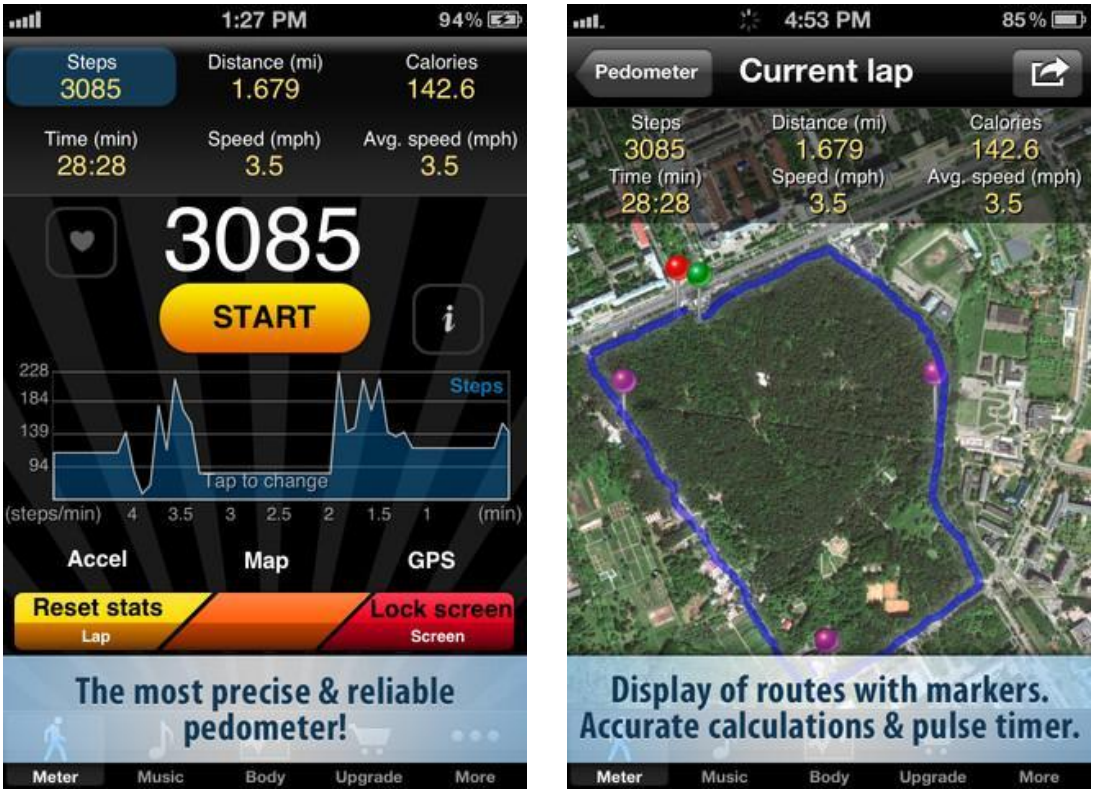


Figure 1[1]: The interface of Free Pedometer by Arawella Corporation

In Figure 1 on the left is a screenshot of the main interface of Free Pedometer by Arawella Corporation. It shows that this pedometer records steps, walking speed and average walking speed. This app also offers the options to switch to map view, reset stats and lock screen.

In Figure 1 on the right is a map view in Free Pedometer by Arawella Corporation. It displays a route the user has walked with a blue tracking line on the screen. In addition, it displays various collected data (steps, distance, calories burned, walking time, speed and average walking speed) for the user.

[1] http://a4.mzstatic.com/us/r1000/076/Purple/v4/31/49/da/3149da32-42d7-6bf8-c41e-20de5461a443/mza_6382548708398678947.320x480-75.jpg

According to their web page, they have more than 5,000,000 users over the globe[2].

It can be deduced from reviews of modern pedometer apps written by enthusiastic users that people use pedometer apps as an exercise tool that promotes health. The main purpose is to use it as an exercise measurement device.

Some product review of pedometer apps are as follows:

> I found it to be a very useful exercise tool[3].
>
> ---- M_Brown984, user of Free Pedometer by Arawella Corporation.

> This app is awesome. I wanted to know how much walking I did at work. I did not realize how much I walked at home[4].
>
> ---- Superman, user of Accupedo

> It's perfect when I walk during breaks and lunch. Have to get those 10,000 steps in[5]!
>
> ---- Angel, user of Accupedo

[2] http://itunes.apple.com/us/app/pedometer-free/id362306160?mt=8

[3] http://itunes.apple.com/us/app/pedometer-free/id362306160?mt=8

[4] https://play.google.com/store/apps/details?id=com.corusen.accupedo.te&feature=search_result

[5] https://play.google.com/store/apps/details?id=com.corusen.accupedo.te&feature=search_result

The product reviews showed that there are various motivations behind using a pedometer app. Some users like to get exercise and get fit, some want to see how much he or she walks every day, and some are happy to set goals and accomplish them.

With this background information, the team made a game that allows users to keep track of their walking distance, acquire points from daily walking and accomplish missions. With features of alliances and coaches, *World Wide Walk* is intended to provide a friendly atmosphere that makes walking fun.

2.2 Related Work

This section discusses prior pedometer games that were references while creating *World Wide Walk*.

2.2.1 Foursquare

Foursquare is a location-based social-networking mobile application. It was created by Dennis Crowley and Naveen Selvadurai in 2009 and gained 20 million users by 2011. It was released on multiple platforms including Android and iPhone. Foursquare has a point award system that operates on the basis of visiting venues to “check-in”. A check-in shows a user’s specific location in a venue, and each user who performed a check-in is awarded with points. The ownership feature in Foursquare is called “Mayorship”, which also awards points. A user gains the Mayorship of a venue by checking-in to that venue more than anyone else in the past 60 days.

Besides giving users the perks of Mayorship and points, Foursquare’s game features are designed around the concept of “social network sharing”. Anyone can access and share other users’ to-do lists, which are lists of events, tips and fun things to do at a venue. Tip exchanging

and meeting new friends are main attractions in this app. However, Foursquare is susceptible to cheating as users can simply check-in to any venue at their home.

2.2.2 Pokewalker

Another pedometer game that was studied was the Pokewalker, a standalone tie-in to the popular game series Pokemon. The device was developed by Nintendo and bundled with the Pokemon HeartGold and SoulSilver games for the Nintendo DS handheld system. It was released in North America in early 2010. The device contains a small screen and three buttons for user input, an accelerometer used to create a pedometer, and an IR receiver/transmitter to allow communication between the Pokewalker and the Nintendo DS system. Users could take one of their pets from HeartGold or SoulSilver, and transmit it to the Pokewalker. Once stored on the device, users could then earn in-game experience for their pet based on how much they walked. Using the buttons and the screen, they could also play mini-games with their pet, earn in-game items, and even interact with other Pokewalkers for more in-game rewards. Nintendo did cap the rewards they could earn; Pokemon in the device could not gain more experience than what was sufficient to advance a single level, and only three in-game items could be held at one time.

Like many other pedometer games that were examined, the Pokewalker shared design flaws that ultimately undermined its effectiveness. By tying the reward system directly to the number of steps taken, the user's incentive was to get as many steps as possible in as short a time as possible. However, because the Pokewalker's pedometer was ultimately unable to detect many forms of cheating, users could achieve the maximum rewards via cheating at comparable rates to normal walking. Because the rewards were only tied into the number of steps taken, users ultimately had no incentive to actually walk. However, despite design flaws, the Pokewalker was

generally well-received. Users enjoyed the fusion of the real world and their gaming world, even though the system could be cheated easily. If an application could be developed without a step-based reward system that achieved a similar fusion of gaming and real-life, then it is conceivable that users would legitimately enjoy walking and exercising as part of a game.

2.2.3 Striiv Smart Pedometer

Striiv is a stand-alone pedometer released in North America in early 2012. Striiv Smart Pedometer observes its users' routines and provides personalized goals to motivate them. It also combines usual pedometer functions with mini games to make walking more fun. An example of its mini game is shown in Figure 2.

[6]

Figure 2: Screenshot of the mini game in Striiv Smart Pedometer

The mini game shown in Figure 2 is called *MyLand*, which turns the Striiv into an island building simulation game. As users walk, run, and climb, they earn energy points that can be spent on

[6] http://the-gadgeteer.com/2012/05/07/striiv-smart-pedometer-review/

coins to buy plants, trees and buildings. The *World Wide Walk* team felt that such a mini game lacks a connection between the game world and reality because the positions of users' activities are not reflected in the game.

Striiv Smart Pedometer also has a feature which lets users donate to charities as they walk. The more users walk, the more sponsors donate. However, all these features still suffer from the accuracy problem of pedometers, as shown in some customer reviews from Amazon.com:

> "Unfortunately, after about 2 months of owning the Strivvs, things starting going downhill. My girlfriends would sometimes count steps, and at times not."[7]
>
> --- R. Trochanowski, September 6, 2012

> "100 steps walking with it in my sweat pants pocket registered 106.
>
> 100 steps walking with it in my jacket pocket registered 107.
>
> 100 steps walking with it hanging from my jacket zipper registered 100.
>
> 200 steps jogging with it hanging from my jacket zipper registered 171.
>
> 100 steps jogging with it in my sweat pants pocket registered 29."[8]
>
> --- G. Evans, January 7, 2012

The Striiv Smart Pedometer is an overall good product. However, if the pedometer games have a stronger connection with the real world, users can feel more immersion into the game. If the

[7] http://www.amazon.com/Striiv-Smart-Pedometer-Basic-Bundle/product-reviews/B00546T5MY/ref=cm_cr_dp_synop?ie=UTF8&showViewpoints=0&sortBy=bySubmissionDateDescending#R3V3O1S6M2UC3F

[8] http://www.amazon.com/Striiv-Smart-Pedometer-Basic-Bundle/product-reviews/B00546T5MY/ref=cm_cr_dp_synop?ie=UTF8&showViewpoints=0&sortBy=bySubmissionDateDescending#R3V3O1S6M2UC3F

instability problem of pedometer can be reduced by combining GPS functions, users can have a better walking experience.

3 Game Design

This section shows an overview of the game design in creating *World Wide Walk*. The audience *World Wide Walk* intends to target, the reasons for choosing Android, and high level concepts on how the game is played will be explained here.

3.1 Audience

The target audience of *World Wide Walk* is people who have an interest in interacting with others through their phones and walking for fitness. The user is required to own an Android smartphone and have an interest in playing games on it. Figure 3 displays a portion of a diagram that shows statistics of smartphone users around the world.

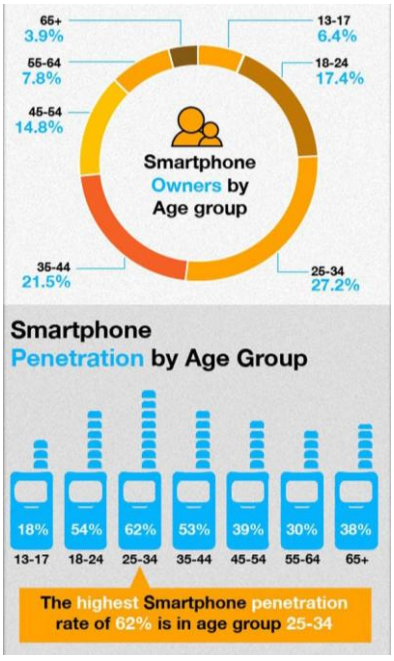


Figure 3: Statistic on Smartphone Users, published by Go-Gulf.com Web Design Company, 2012-01-02 [9]

[9] http://www.go-gulf.com/blog/smartphone

As shown in the Figure 3, the most active smartphone users lie in the 25-34 age range. Twenty seven point two percent (highest) of all smartphone users lie in the 25-34 age range. They also have the highest smartphone penetration rate of sixty two percent. The idea of *World Wide Walk* is to provide a healthy, friendly, and entertaining game environment for a wide range of people, not only for the major 25-34 age range smartphone users, but also for teenagers and older users. On the artistic side, the team decided to include both Eastern and Western oriented visuals in order to appeal to audiences with different preferences. The team also created a flyer to promote this project at the early stage (included in Appendix B). The flyer was made to gain people's attention and support for this project.

3.2 Overview

World Wide Walk is a game that is intended to be accessible for users to play. Ideally users do not have to devote extra time to the application. As long as the application is turned on and users are walking outdoors, points are automatically accumulated. Figure 4 gives an overview of how the game works. For screenshots of each section, please refer to Section 4.3 User Interface.

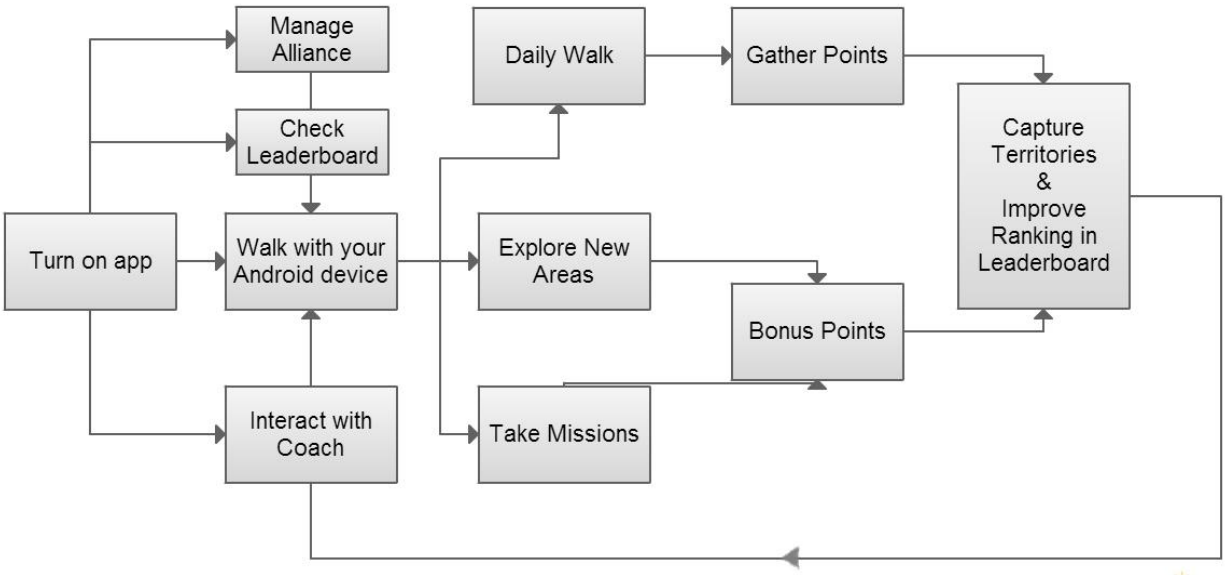


Figure 4: Gameplay flowchart

As shown in Figure 4, once users turn on *World Wide Walk*, they have the ability to join or quit an alliance, check the leaderboard, interact with their coach, or start walking. Alliance management includes the ability to join or quit an alliance, create a new alliance, dismiss alliance members or dismiss an alliance as the creator of that alliance. Leaderboard shows the user where he/she ranks in the global leaderboard. Coach interaction includes mission offers, various reminders and daily greetings. All these functions can be accessed at any time.

Once users start walking with their devices, they start to accumulate points and if users walk into a new area, bonus points are awarded (please refer to Section 5.7 Point System for details). Users can also accept missions that are offered by their coach. Missions require users to accumulate a certain numbers of points within limited time. Bonus points are awarded if users complete the missions.

The goal for a *World Wide Walk* user is to gather as many points as possible. The more points a user gets, the higher they rank in the leaderboard. In addition, if a user gathers the most points

on a street, they can capture that street. However, any street ownership is not permanent and other users can take over if they accumulate more points than the current owner. Users have to keep walking to secure their street ownerships. When a user owns enough streets in a territory, which is a pre-defined area by the application, he/she is said to own the whole territory. There is no limit of how many points a user can gather in *World Wide Walk* and therefore there is no end to this game. Users can walk with *World Wide Walk* as long as they want to.

3.3 Platform

In the beginning, the team debated on creating a game on each of the different platforms, from PC to game consoles to mobile devices. The choices were then narrowed to Android or the iPhone. The resulting choices were between iPhone and Android, and the two platforms were compared. Having only two programmers on the team and knowing they would be distracted by the different environment of Japan greatly influenced the decision.

Android apps are written in the Java language whereas iPhone apps are written in Apple's Objective-C. Java was the easier option for two programmers more proficient at it than learning an entirely new programming language.

The learning curve between Android and iPhone differs greatly. For the Android, the support system is through creating sample apps that are easy to learn from. One can download many open-source Android projects and learn about the architecture and API usage. For the iPhone, the learning process mostly include videos which makes learning and understanding important concepts easier, but the video progresses slowly and finding the required information would take hours instead of minutes.

Android provides an open development system whereas Apple is more closed. Having an open system can allow developers to innovate new game mechanics while being able to display multiple notifications. The downside is that Android developers have to understand and handle multitasking on the Android platform and deal gracefully with notification interruptions like receiving phone calls. On the other hand, having a closed system means the structure, framework, and hardware is well-defined and known to the developers. With such a closed system, users expectations are met with minimal fuss on getting the app on the phone, what the app does when on the phone, and what features the app will support.

Android also has an easy development environment and testing tools. Android uses many Java Development Tools (JDT) tools that are standard tools in every Eclipse installation. The integrated development environment (IDE) for Apple, Xcode, can only be run on a Mac computer and it's UI designer is more developer friendly than the one integrated in Eclipse.

There were many pros and cons between the two platforms, but three points stood out in the comparisons. One point was Java vs. Objective-C in Android and iPhone respectively. The two programmers on the team felt that learning an entirely new language on a platform they have never worked on to create an app in two months was not feasible. Programming in Java on the Android would increase the feasibility of creating an app in two months. The second point was Open vs. Closed System in Android and iPhone respectively. At the time it was not known exactly what kind of system the application would need. The open system seemed to hold more options for developers and expand the creativity as opposed to the closed system. The third point was the IDE choice. Setting up the IDE for Android would have been easy because Eclipse contained the necessary JDT tools. Development in iPhone though would require a Mac which no one on the team owned. These three points made Android the better choice.

3.4 Game Mechanics

World Wide Walk's mechanism consists of walking, points, missions, territory ownerships, alliances, coaches, and leaderboards.

In *World Wide Walk* points are the users' scores. Users walk in the physical world with *World Wide Walk* running on their Android device to gain points. Points earned are determined by walking distance, walking speed and the slope of the ground. Then the sum of points total is updated in the device database. Please refer to Section 5.7 Point System for formulas on how points are calculated.

The game rewards users with bonus points in two ways; completing missions from coaches and whenever users walk in new areas. These bonuses are a small portion compared to consistent walking because users who make walking a daily routine should not feel their efforts are ineffective in the game. In fact, most people walk similar routes in their daily life, either to work or to school.

Missions are intended for users who need incentive or encouragement to walk. At random intervals, users will receive a notification that their coach has a mission for them. If the user chooses to accept, the coach will then declare how many points they must gain through walking. Once the mission is complete, users are awarded bonus points for their efforts.

The virtual globe in *World Wide Walk* is divided into many small areas known as territories. These territories can span neighborhoods, cities, parks, anything and anywhere users can walk. Whenever a user earns points within a territory, their total score rises and those points are associated within that territory. The users who holds the most points in a territory then owns that

territory and the territory's color changes to the color the user chooses which is reflected on everybody's map. But ownership takes work to maintain. Each week, effective point-totals for each territory are reduced by ten percent, thus making old points less effective on territory claim. Users must continually walk and earn new points if they want to keep their ownership over territories. This was intended so that users would have more motivation to walk.

Further motivation to walk comes from the leaderboard. The game takes users' point-totals and ranks the users from most to least number of points. Users will see the top users in the leaderboard staking claim over the world of *World Wide Walk*. Users who wish to see their rank climb will then have to walk even more.

The alliance system is designed for users who want to connect with other walkers. Users can join alliances in *World Wide Walk* and make friends with those in their neighborhood or across the globe. An alliance can capture territories as well, but is separate from individual territory ownership. The feeling of contributing to a group is an important aspect to the social side of *World Wide Walk*. All users would receive notifications whenever a user joins or leaves the alliance.

4 Art Development

This section discusses the design of the art contents of *World Wide Walk*. The choice of the art style, user interface design, general coach design and detailed information for each coach will be covered.

4.1 Artistic Vision

The artistic vision of *World Wide Walk* was to present the game in an entertaining environment. The game has a lot of light colors going in contrast with a few dark colors to convey the cheerfulness of the game. The art team considered two main aspects when deciding the style of coaches: screen resolution and audience. In order to make images clear while showing on a small portion of the screen, the artists decided to use a cartoony style. The artists also decided to represent the coaches as half-length portraits due to the limited space on a phone screen, in the same way as many Japanese Role Playing Games show characters (as shown in Figure 6).

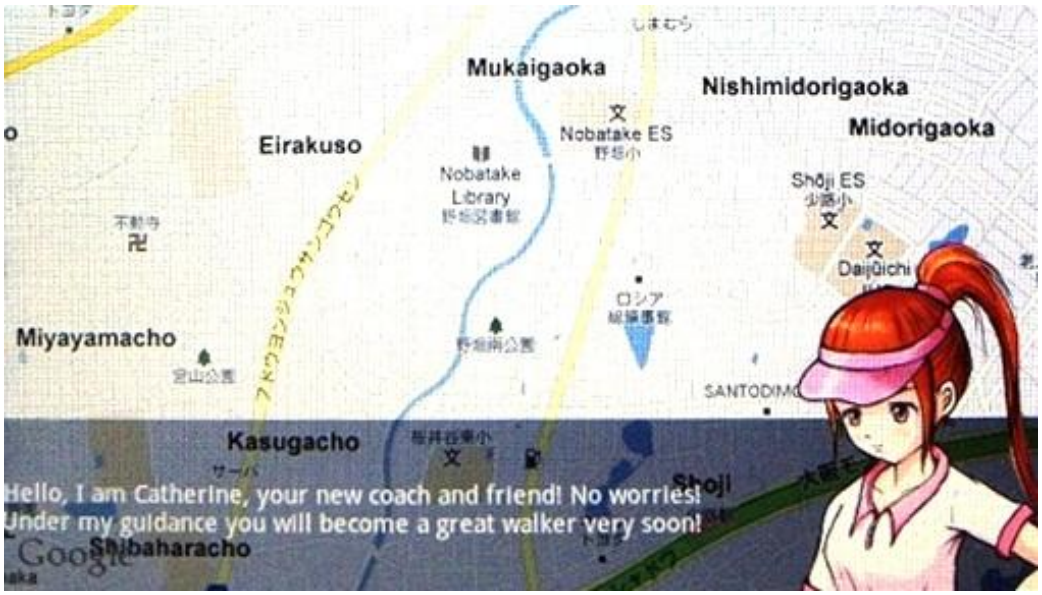


Figure 5: *World Wide Walk* screenshot showing the portrait of the coach

As shown in Figure 5, the coach only takes up a little space on the screen but still can be seen clearly. While this is the general direction of design, each coach was designed differently.

4.2 Western and Eastern Styles

There are two sets of skins in *World Wide Walk* represented by two styles: Western style and Eastern style. Each skin consists of six coaches and individual UI elements. Users are free to choose either one for the game. The making of two skins was a process to bring each artist's strength to the game. This MQP was conducted in Japan. The team was inspired by the Japanese culture and wanted to integrate their appreciation into this game. In addition, the team came from a Western institution, which has a different aesthetic. The artist taking the role of creating the Western skin appreciated her education in the west and volunteered to contribute her skills to the game. A lot of effort has been made by the art team to keep both styles unique and consistent. More importantly, the team expected that some users might prefer Western

style and some might prefer Eastern style. By having both styles in the game, *World Wide Walk* can visually appeal to a large range of audience.

4.2.1 Western Style UI

Figure 6 and Figure 7 below shows the coach button and the coach selection screen in Western style skin.

Figure 6: Screen shot of Western style UI - Coach Button

UI elements in Western style are clean and simple, just like the Western coach designs. Buttons include generic shoe and paper icons which were drawn with bright colors and black borders. Menu selection buttons are crystal blue with shine.

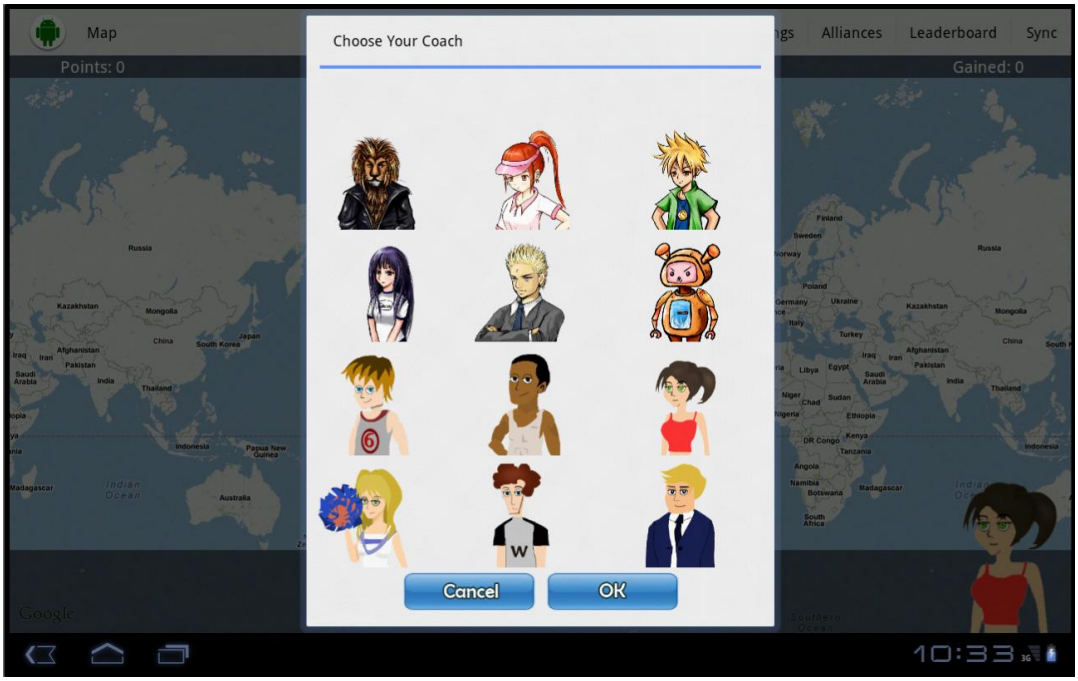


Figure 7: Screen shot of Western style UI - Coach Selection

4.2.2 Eastern Style UI

Figure 8 and Figure 9 below shows the coach button and the coach selection screen in Eastern style skin.

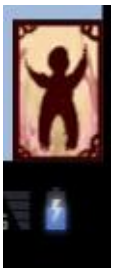


Figure 8: Screenshot of Eastern UI – Coach Button

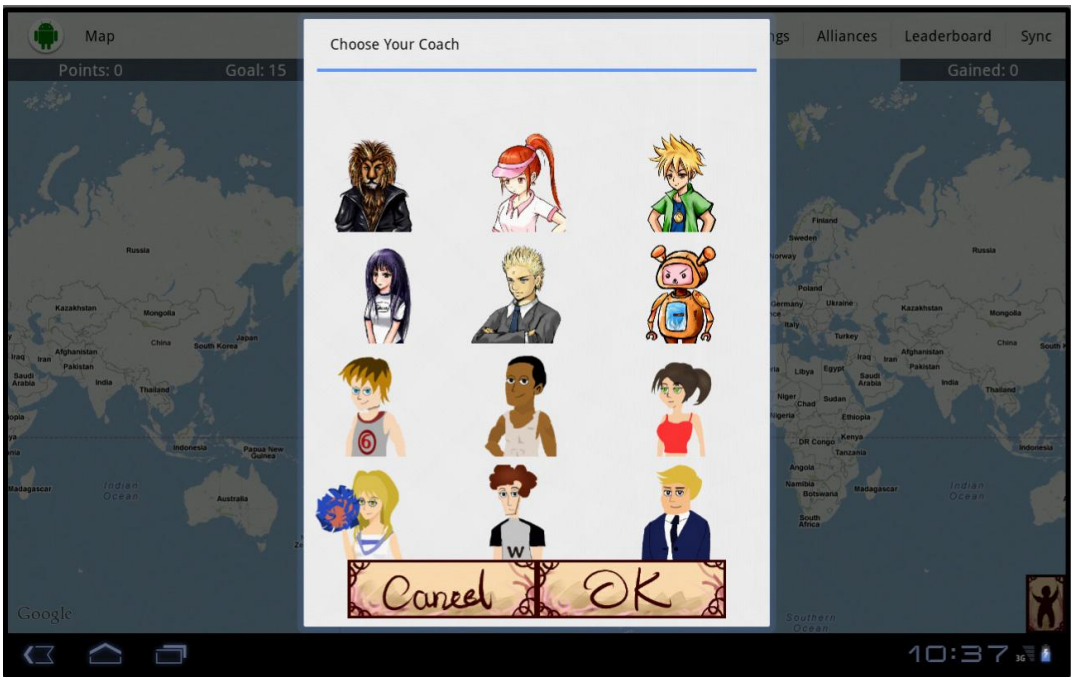


Figure 9: Screenshot of Eastern UI - Coach Selection

UI elements in the Eastern style are inspired by traditional Chinese and Japanese decorations. The background of the buttons resembles antique Chinese screens. The texts on the buttons are hand written to achieve a calligraphy effect.

4.3 User Interface

The user interface is the key feature connecting all the background coding to a tangible resource on the users' screen. It also connects the artist's work to the application for users to interact and use. *World Wide Walk* begins with a startup screen containing a globe-shaped logo and a "Tap to Start" ImageView acting as a button, as shown in Figure 10 below. This was so the users were not immediately engaged in the game and allowed the program to determine if the user had used the application before or not.

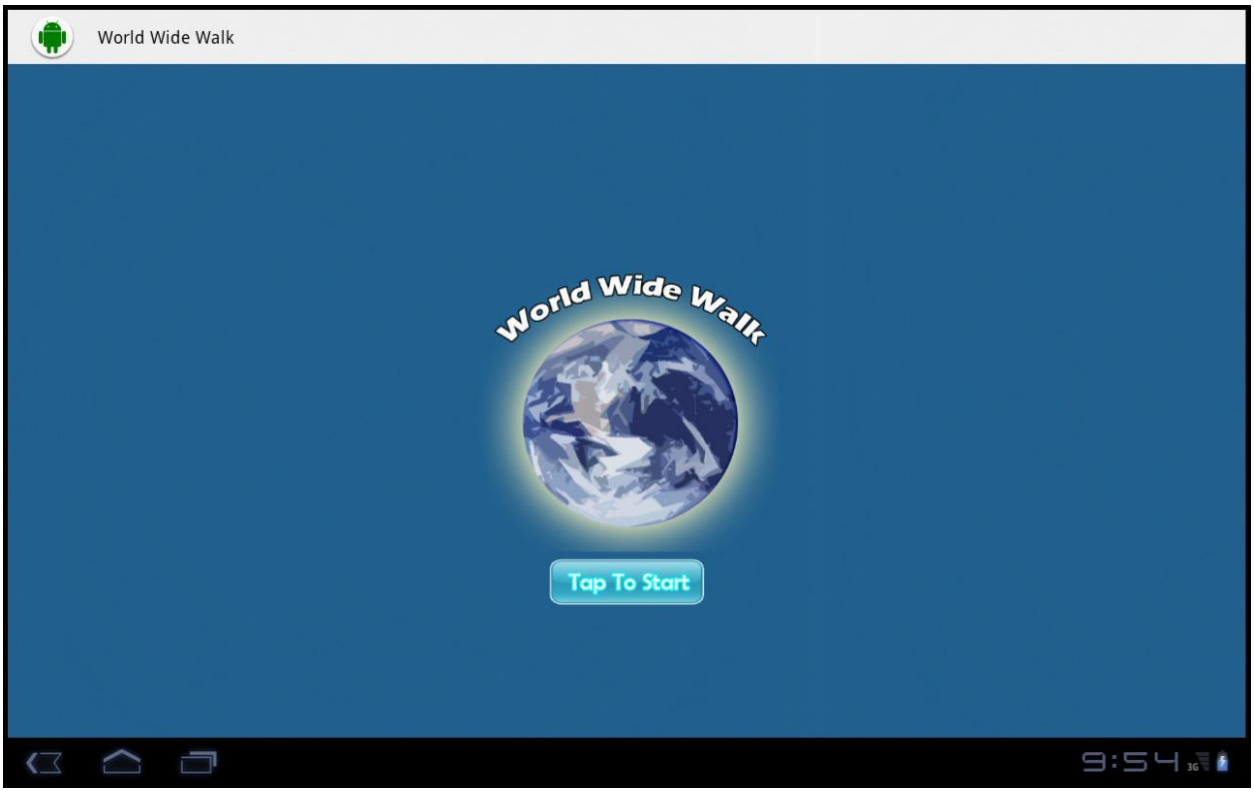


Figure 10: Screenshot of Startup Screen

Figure 10 is a screenshot of startup screen with the logo of *World Wide Walk* and the "Tap To Start" button

When the users press "Tap to Start" ImageView, the program starts a new activity on top of the startup screen. This second screen is where the users begin playing *World Wide Walk*. If it is the users' first time, they are instructed to input a valid username while they play by using an AlertDialog (Figure 11). After the user has entered a username, the AlertDialog box is replaced with a Dialog box which contains images of the twelve coaches the users can choose from (Figure 12).

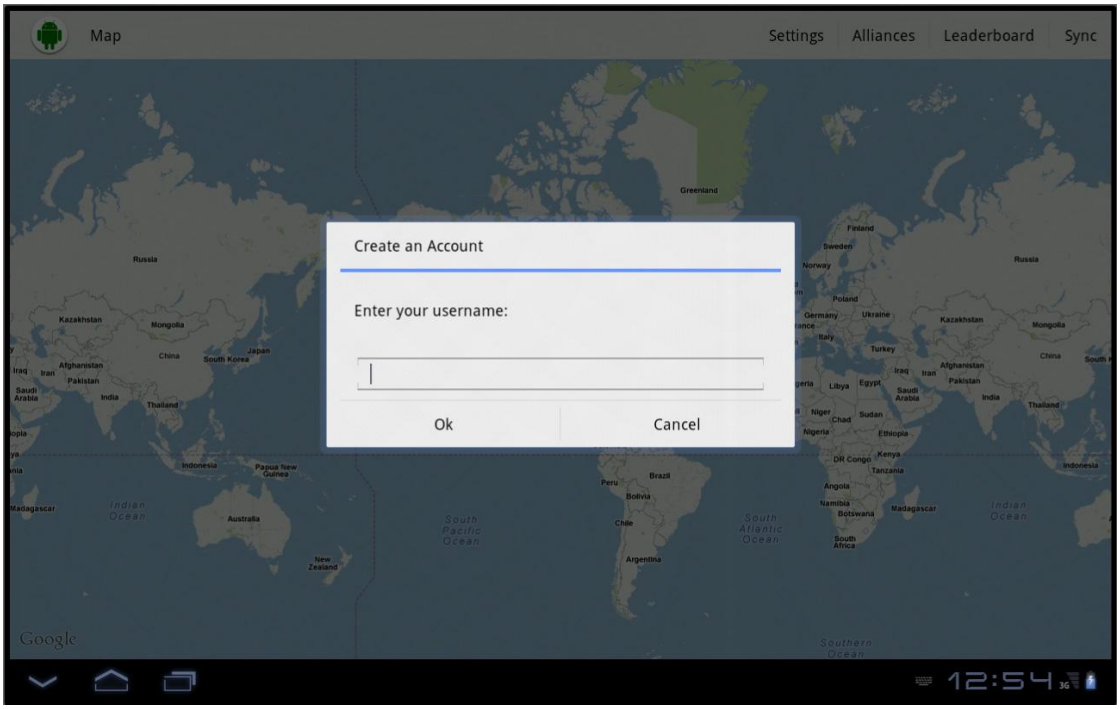


Figure 11: Username input

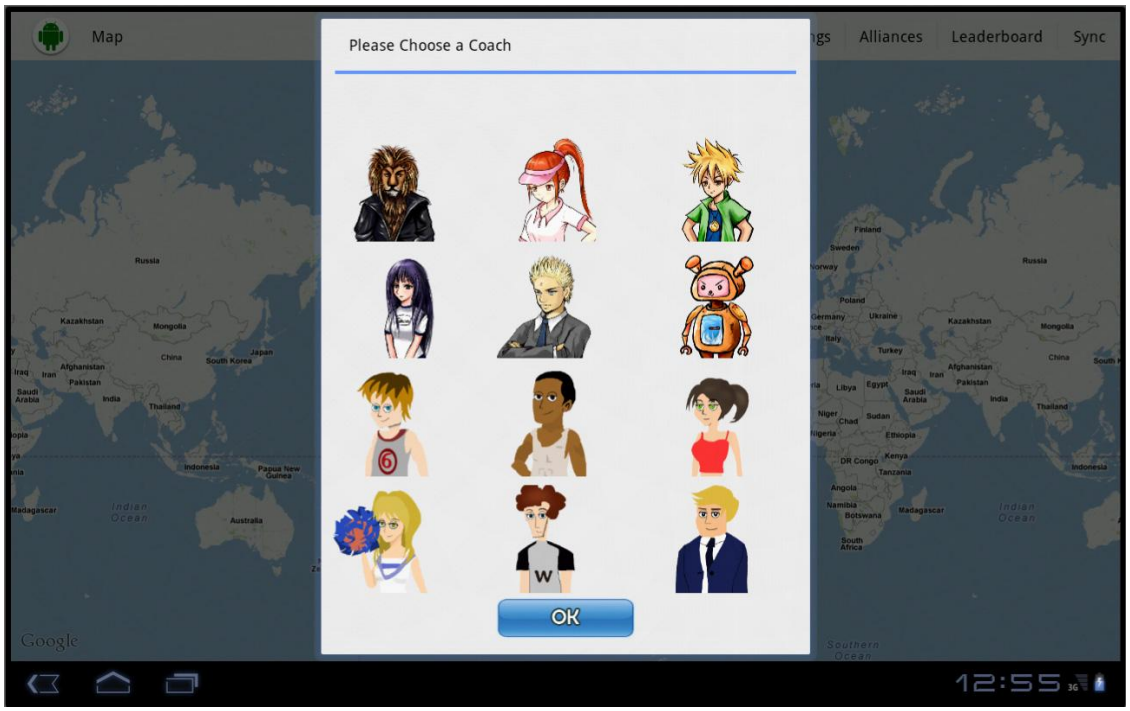


Figure 12: Coach Selection

Once the coach is picked, the user is brought to the game screen where the map is shown and, if the location provider has discovered the users' location, zoomed-in to the users' position through a method provided by GoogleMaps. The total amount of points the user has is shown in the upper left-hand corner of the screen on top of the map. At the bottom, the coaches appear when certain conditions are met. When the users are given goals to reach, the goal points are centered along the top of the screen and the points gained to reach that goal is shown on the upper right hand corner as shown in Figure 13. These two UI components disappear when the goal has been met.

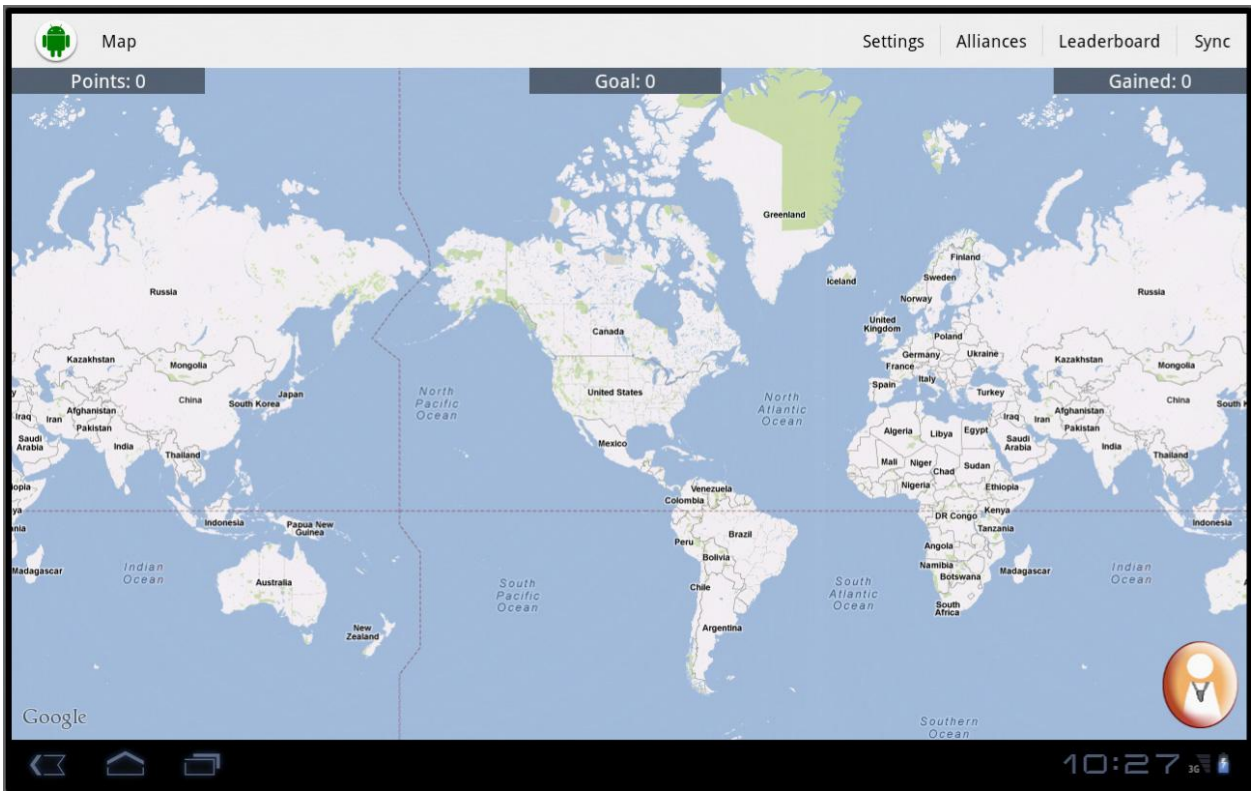


Figure 13: The default game layout

Along the top of the screen next to the application icon, is the menu bar where the users can change their settings shown in Figure 14. Through the settings menus, users can change their coaches. They can change the method of notification when points are gained in Figure 15. And they can change the color of their owned territory in Figure 16.

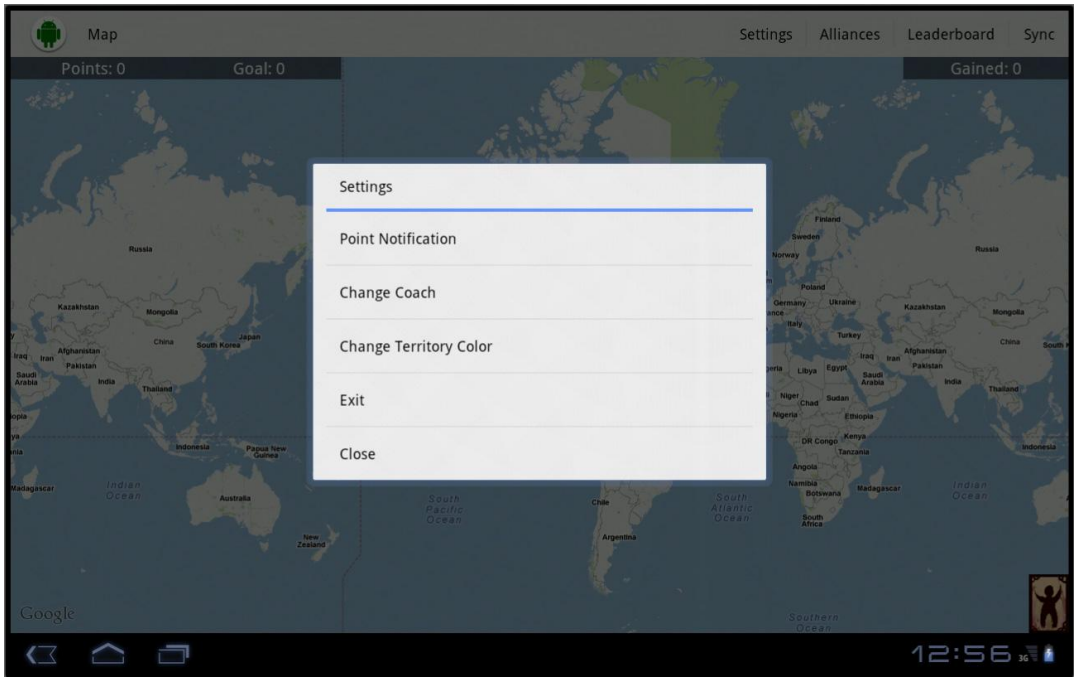


Figure 14: Settings

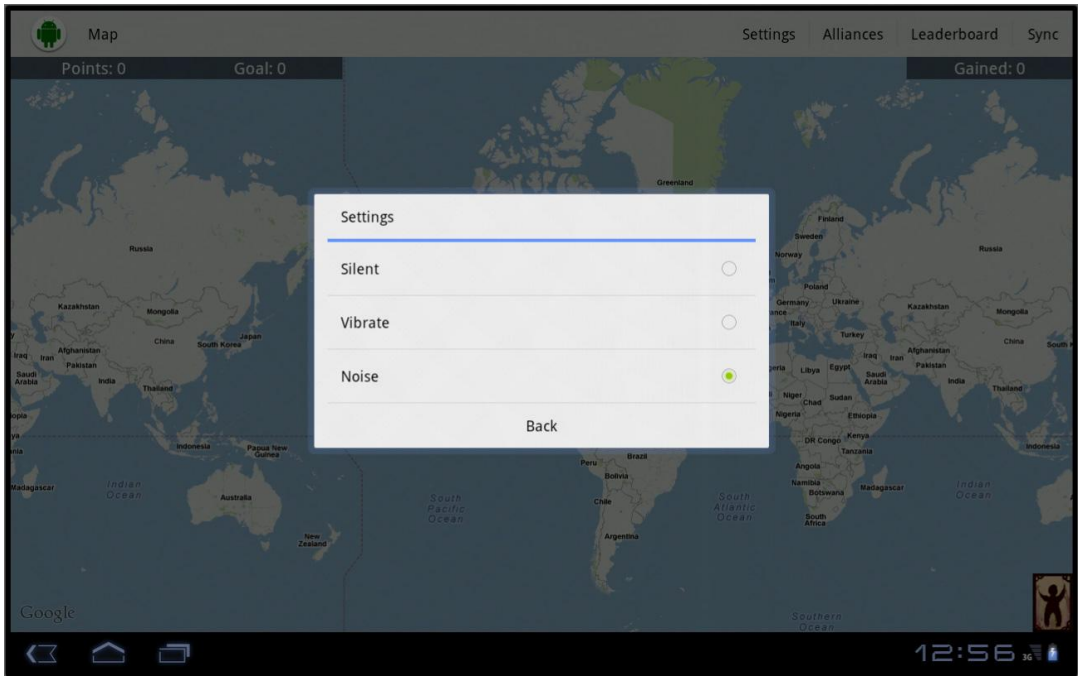


Figure 15: Choose method of notification when points are gained

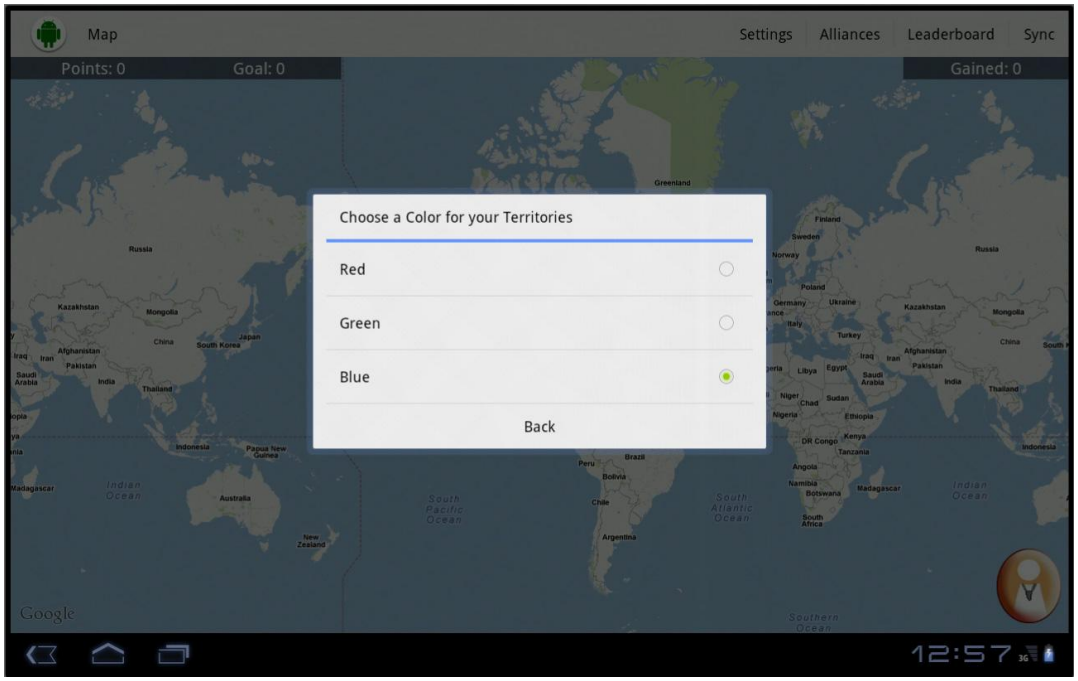


Figure 16: Choose color for territories

The menu bar also contains the ability to join, start, manage, or leave an alliance shown in Figure 17-21.

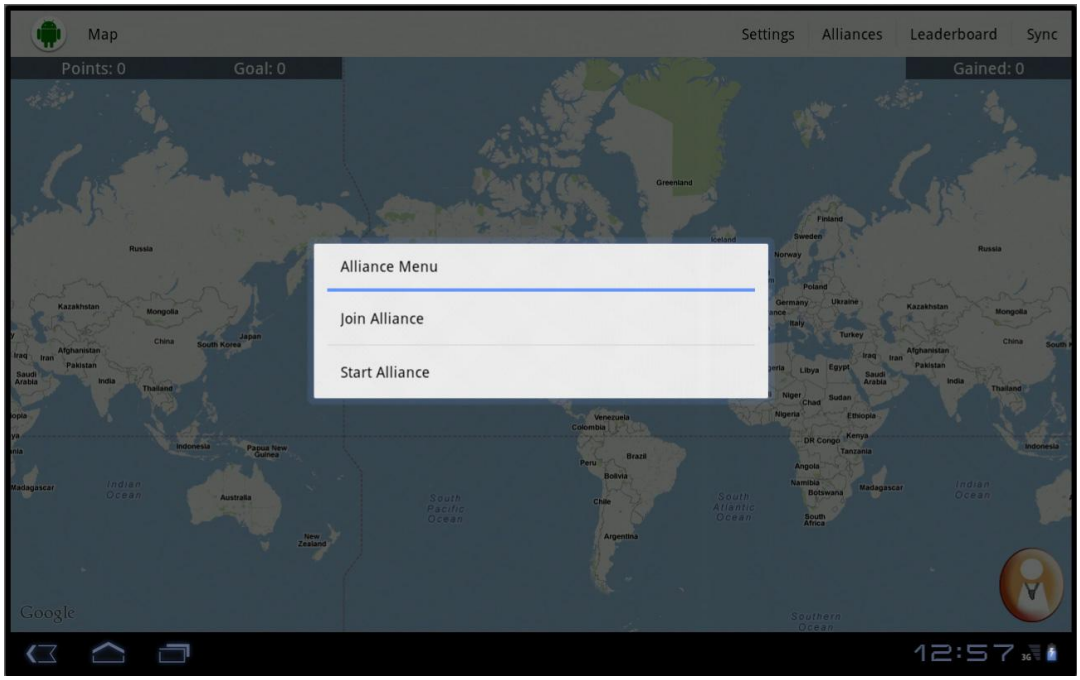


Figure 17: Alliance Main Menu

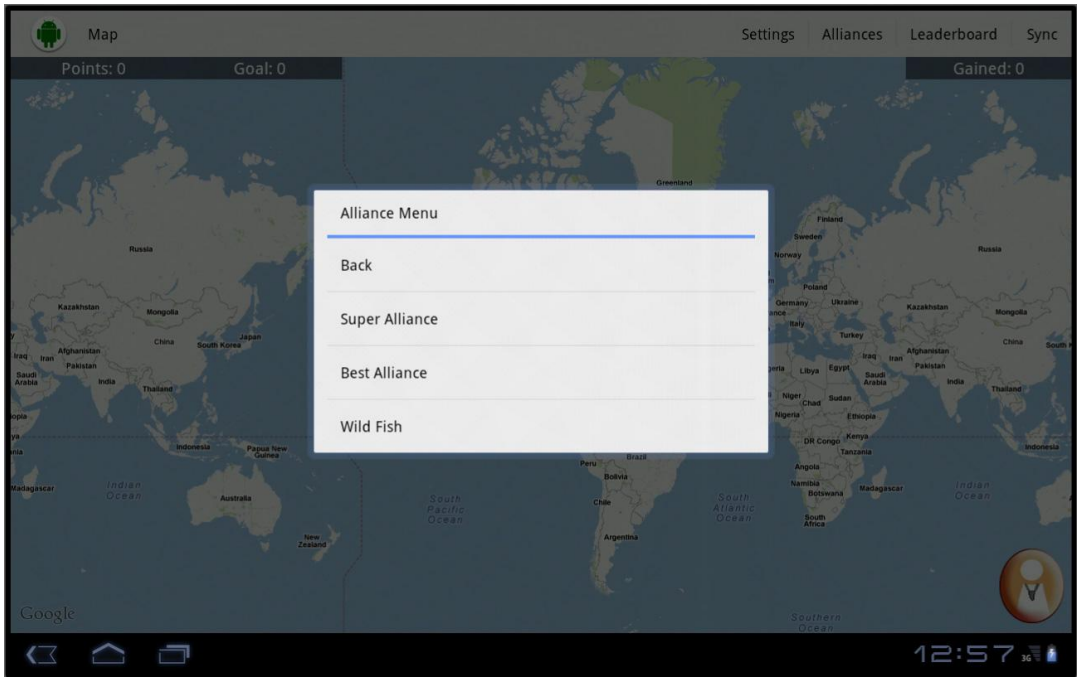


Figure 18: Join an Alliance

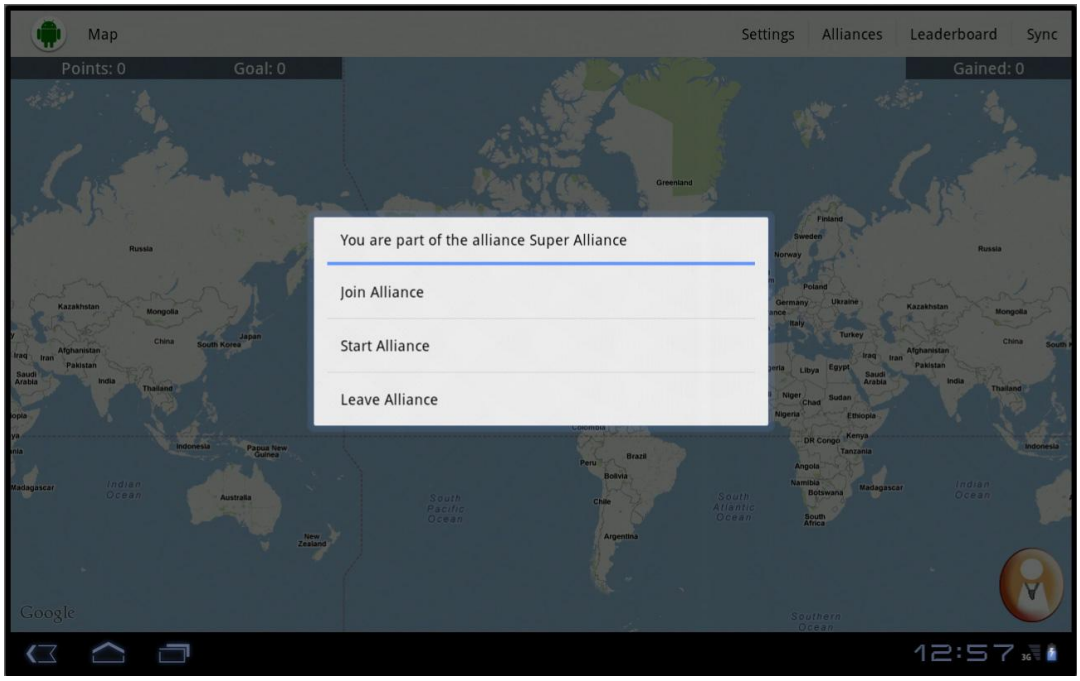


Figure 19: Join an Alliance

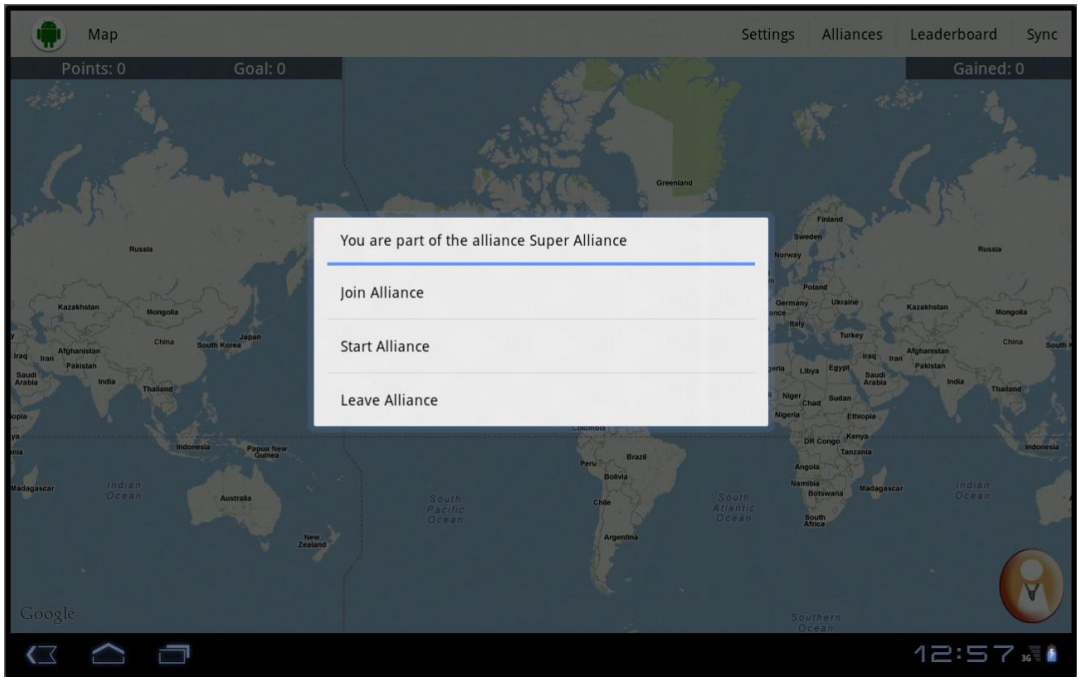


Figure 20: Start an Alliance prompt

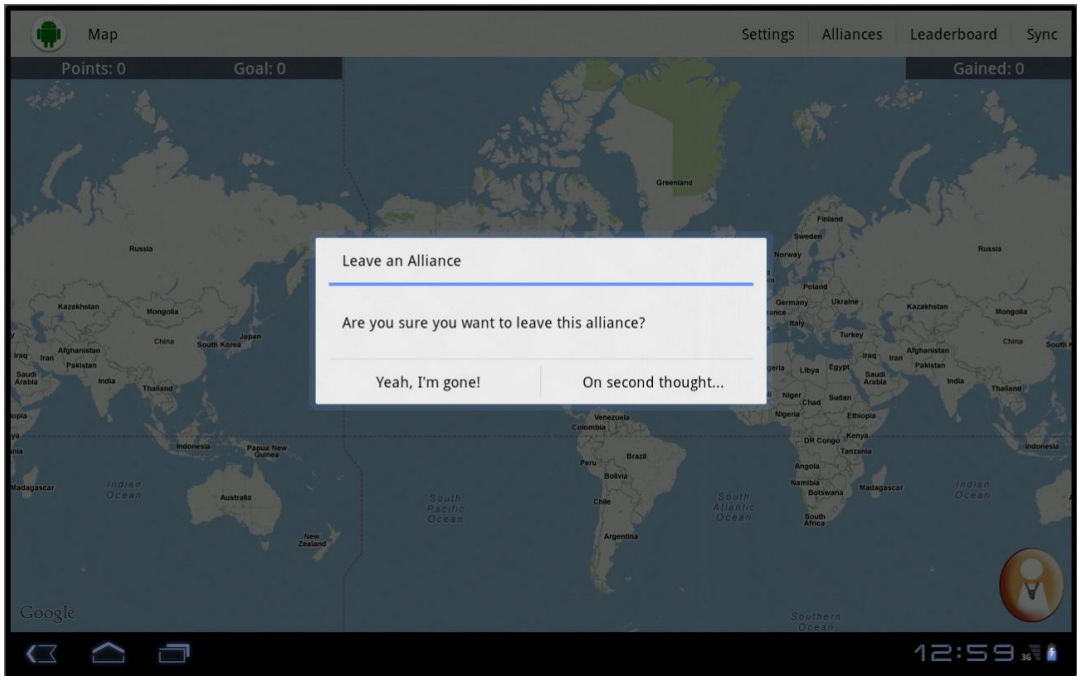


Figure 21: Leave an Alliance

The menu bar also allows users to bring up the leaderboard (an AlertDialog box) shown in Figure 22 and Figure 23, and Sync their information with the central database as seen in Figure 13 in the upper right hand corner.

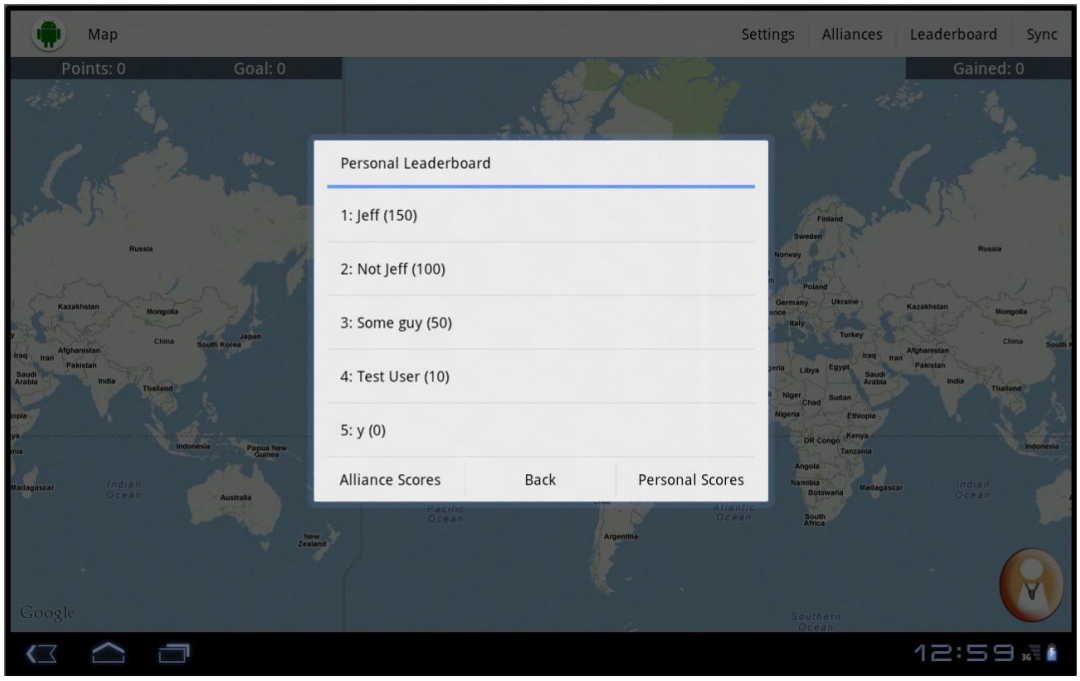


Figure 22: Individual Points Leaderboard

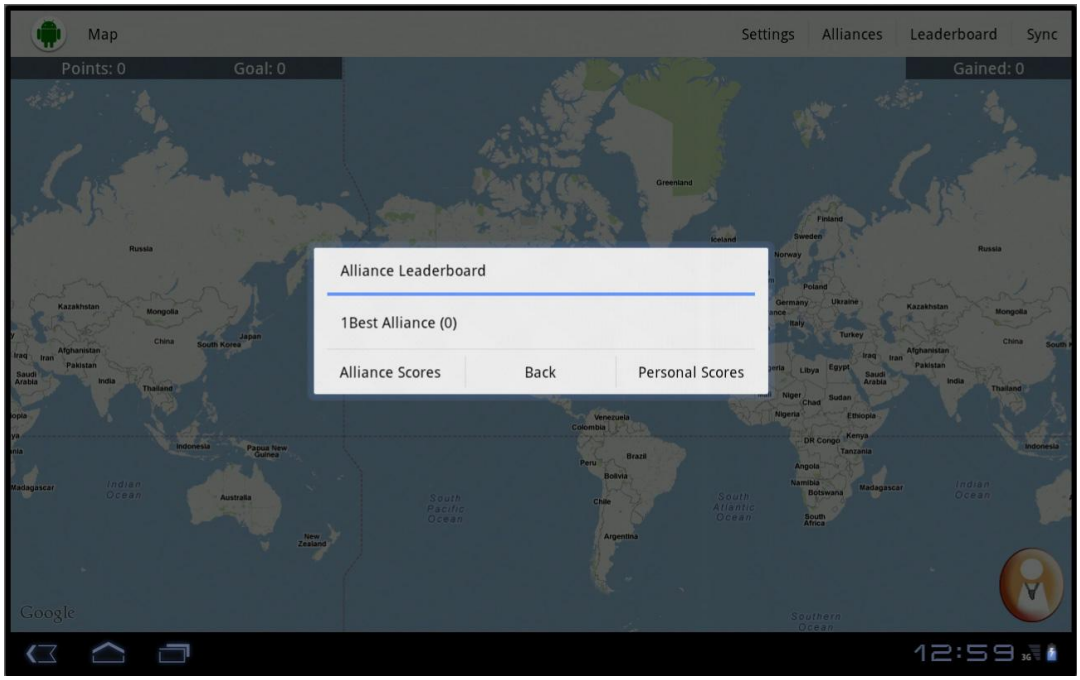


Figure 23: Alliance Leaderboard

Some minimal UI changes are determined by the coaches picked. The styles of the chosen coach determines the coach button, which brings up the coaches, and the coach selection buttons, which can be seen when users want to change their coaches. (See Sections 4.2.1 and 4.2.2)

When users begin playing the game, a red dot can be seen on the map. This shows the location of where the user is currently. As they move, the red dot moves and a red line is drawn on screen as shown in Figure 25. This red line displays the route of where the user has been in for the duration the game has been running. (See Section 5.3 Map & Routing for further explanation) In Figure 24, it is hard to see, but the darker grey are representations of the buildings in the area.

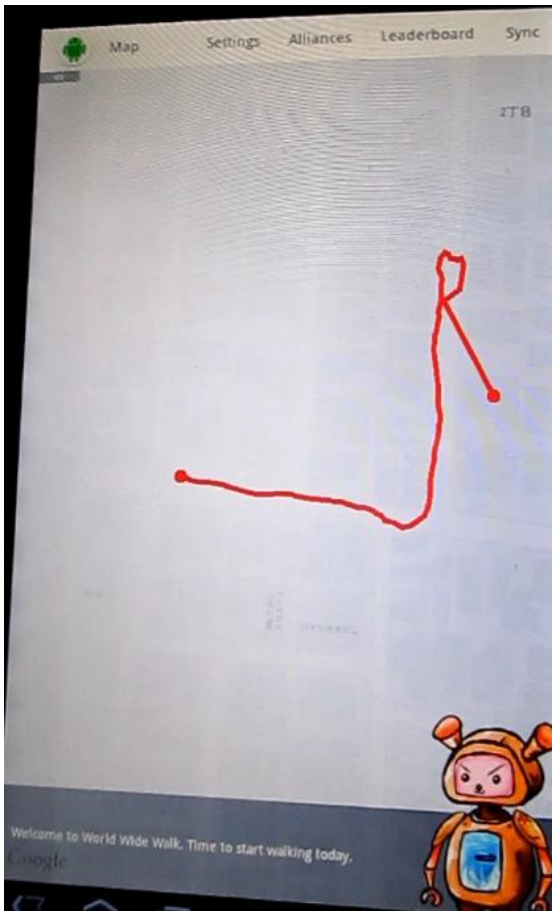


Figure 24: Route displayed on screen

4.4 Coach System Design

Coaches are companions that interact with the user throughout the game. The main reason this was designed in the coach system was so the users of *World Wide Walk* could feel more connected to the game world. Therefore the coaches are designed to communicate with the users in a friendly and entertaining way. Each coach has his/her own personality that makes the game experience different for users with different coaches. The coach watches over the users' activity and provides feedback accordingly. Technically, the coaches are triggered under various conditions which are shown in Table 1 below. Note: "X" & "Y" are variables which are different for different coaches.

Table 1: Coach Notification

Condition	**General Message**
Application start	Daily greeting
First meeting	Greetings and self-intro
No walking in "X" days	Stop being lazy!
Non-stop walking for "Y" hours/mins	Good job! / Shall we take a rest?
Walking speed too fast (in a car, biking, running, etc.)	It doesn't count for our game.
Mission 80% complete	Almost there!
Mission complete	Good job!
Territory captured	Congratulations!
Territory ownership lost	Please work harder
Territory captured(alliance)	Congratulations!
Territory ownership lost(alliance)	Please work harder
New member joined alliance	Cheers!
Member left alliance	What a pity

Table 1 is a template of coach dialogue design. Generic messages correspond to different conditions.

The coaches were designed to be noticeably different from each other in more than just appearance. Specific dialogs were designed for each coach. Also various expressions were drawn for each coach according to his/her personality.

The coaches' different personalities needed to be reflected in the game play. Therefore the team introduced the "X" and "Y" variables in the first two conditions (first two rows on Table 1). The stricter the coach is, the smaller "X" will be and the bigger "Y" will be for that coach (specific "X" & "Y" values for each coach can be found in sections 4.5 and 4.6). For example, a very strict coach (Coach A) will yell at the user for not walking in just one day whereas a compassionate coach (Coach B) might only remind the user after three days. The value of "X" here for coach A is "1", and for coach B is "3". Although each coach has different lines under the same conditions, the general message under each condition is given in the table above as a guideline to specific coach dialog designs. The other dimensions of the coach's personalities are also reflected in the specific dialogs.

From time to time, the coach appears at random intervals with missions that require the user to gather a certain amount of points. It is up to the user to decide whether to accept these missions or not. There is no penalty for rejecting a mission. However, if the user completes the mission, he/she is awarded 50 points.

The artists picked up five essential traits that were deemed important to a coach to make the star trait diagram, as shown in Figure 25.

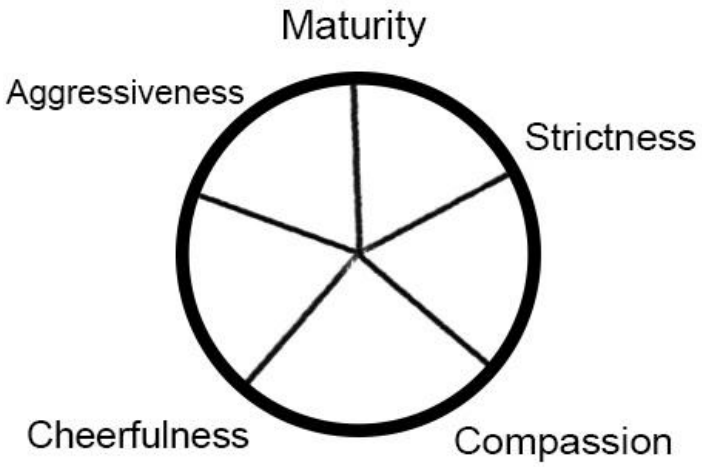


Figure 25: Star Traits for coaches

The reason why the artists chose these five traits was through determining how the coach interacts with the user in the context of the game. A trait that represented how likely the coach would push the user to capture more territories was needed, therefore “aggressiveness” was chosen. A trait that showed how likely the coach would speak and act in a formal and professional manner was needed; therefore “maturity” was chosen. A trait that showed how likely the coach would encourage the user was needed, therefore “cheerfulness” was chosen. A trait that showed how likely the coach would push the user was needed, therefore “strictness” was chosen. Lastly, a trait that showed how much the coach personally cares for the user was needed, therefore “compassion” was chosen. The artists used the diagram as a reference while designing each coach in order to make sure that the star traits for each coach did not overlap too much. The artists decided to create six coaches for each set since there were five traits on the diagram and the artists wanted to have two sets of coaches (one for Eastern style and one for Western style). Each coach would have one dominant trait from the five traits that the artist chose. Thus all the coaches combined would cover all the five traits from the star trait diagram.

4.5 Western Coaches

The Western style is simple and clean. An analogy can be drawn to the style of the popular game Angry Birds. The artist also looked at references like Diner Dash when creating the Western skin, because the Diner Dash series is a successful representation of the art execution of casual games. Figure 26 shows a promotional art of Angry Birds and Figure 27 is a screenshot of Diner Dash: Hometown Hero.

Figure 26: Promotional art of Angry Birds[10]

Figure 27: Screenshot of Diner Dash: Hometown Hero[11]

Above figures illustrate the commonalities in Western casual games' art styles. Both games' art have clean line work and bright colors.

[10] http://t3.gstatic.com/images?q=tbn:ANd9GcTFShRf4lsP6xjEBJQFTgG7vT2zBUAeKcEBChkkVjIc7JiwtZLNag

[11] http://www.29soft.com/download/diner-dash-hometown-hero/

The typical Western casual game style has clear line work and uses large color blocks to represent different parts of a character, as shown in Figure 28:

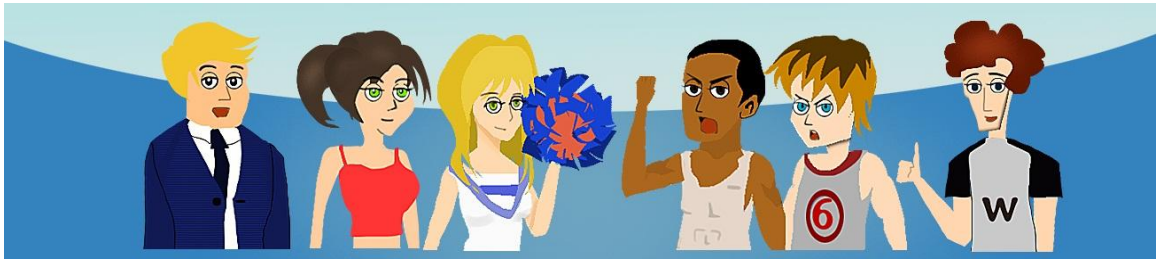


Figure 28: Group shot of Western style coaches

Figure 28 is a screenshot of all six coaches in Western style skin. Each coach has a unique expression and outfit

Following these basic guidelines, the team wanted to show that *World Wide Walk* is a casual game and created the Western skin to bring a friendly environment to the game.

More details on individual coaches are discussed in the following sections.

4.5.1 Draco

Draco is a typical athletic coach. He is somewhat mature for a twenty-six-year-old. He cares about the users' health and therefore is very strict when it comes to walking for fitness. He is happy when he has helped them to meet your goal. On the other hand, Draco is not a compassionate coach and will get aggressive if they don't walk in two days.

Figure 29 below is an expression chart of Draco

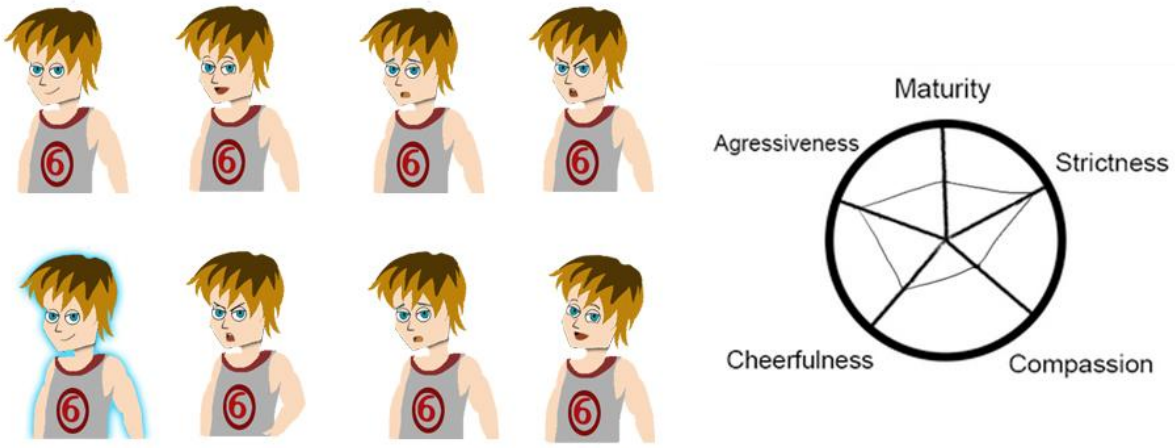
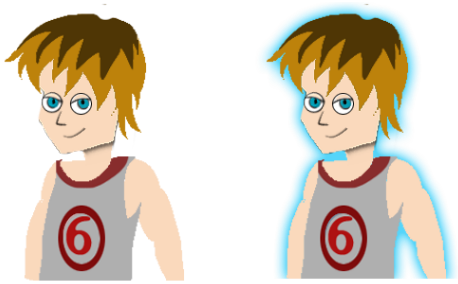
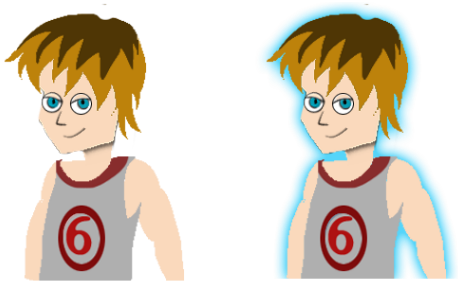
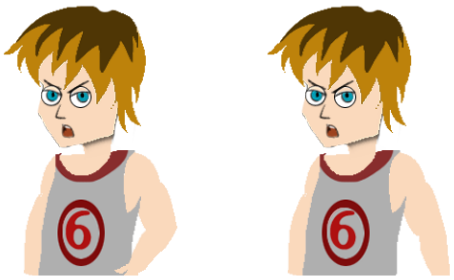
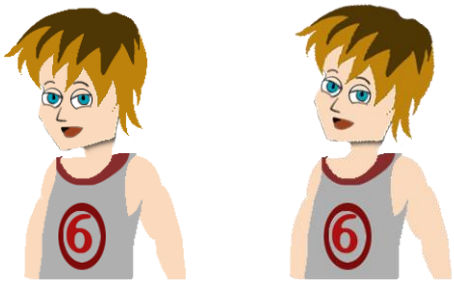


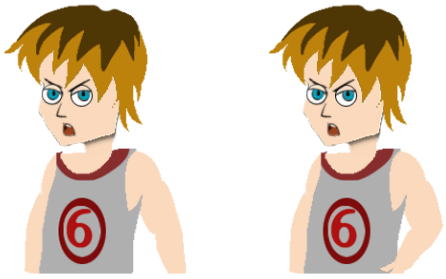
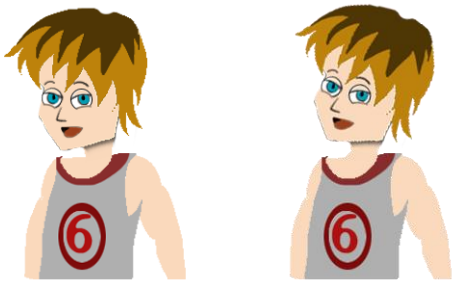
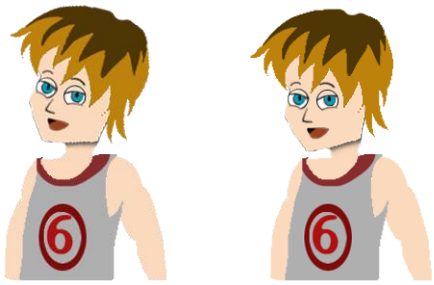
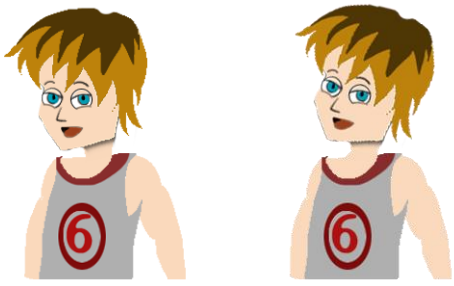
Figure 29: Draco's expression chart

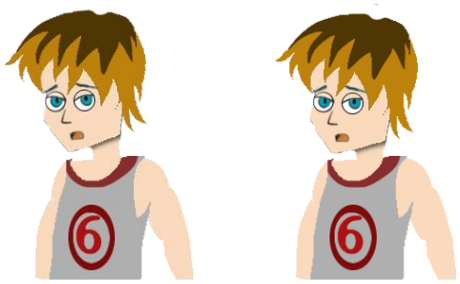
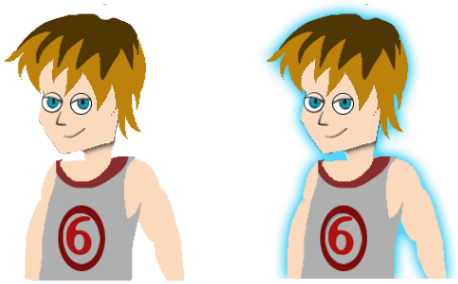
Figure 29 illustrates Draco's expressions and personality. As an athletic coach, Draco wears a simple grey jersey. His brown hair looks energetic in comparison. Draco was number six on his basketball team before he became a coach. He wears his team jersey to remind him how wonderful life can be with sports.

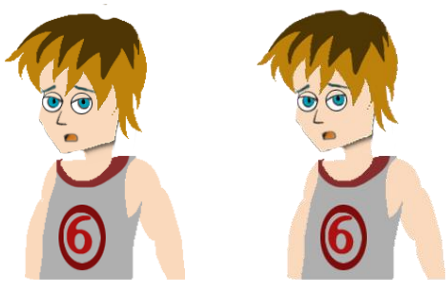
Table 2 below is the complete dialogue design for Draco.

Table 2: Draco's Lines

Condition	General Message	
Application start	Hi, I'm your coach Draco. Walk with me!	
First meeting	Nice to meet you. My name is Draco. I guarantee that you are going to get a fit body under my watch. What do you say?	
No walking in 2 days	You've been missing for two days! We need to get you back on track ASAP!!	
Non-stop walking for 2 hours	I'm proud of you. This is serious walking.	

Walking speed too fast (in a car, biking, running, etc.)	Dude, get OFF that vehicle and start WALKING!	
Mission 80% complete	Hang in there! You are almost there!	
Mission complete	Done! Look into the mirror, you are much more FIT!	
Territory captured	This street is yours! Better make sure it stays that way!!	

Territory ownership lost	What? This street is lost? LAME.	
Territory captured(alliance)	Cool. Let's go out and get more streets!	
Territory ownership lost(alliance)	You've got to work harder to make up for this loss!	
New member joined alliance	Look, there's a new walking machine...	

Member left alliance	Too bad, not everyone can handle the training like you.	

4.5.2 Rick

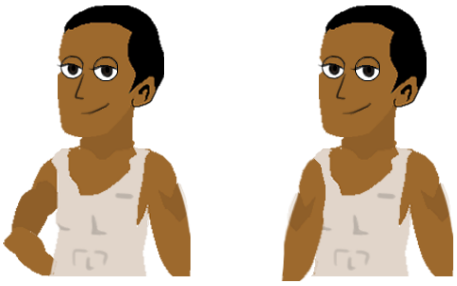
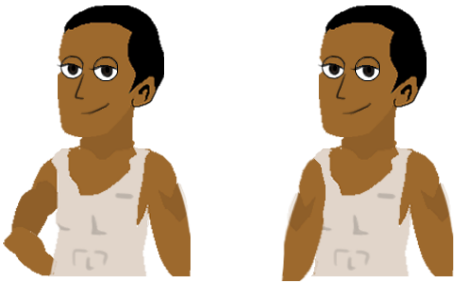
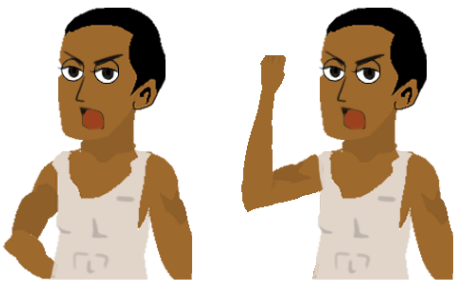
Rick is the sternest of all the coaches in Western style skin. He proactively reminds the user when he or she does not turn on the app and walk every day. Rick sees helping the user as his mission and relentlessly gives users hard training. However, Rick is a mature and compassionate man. He understands if one wants to take a break. Figure 30 below is an expression chart of Rick

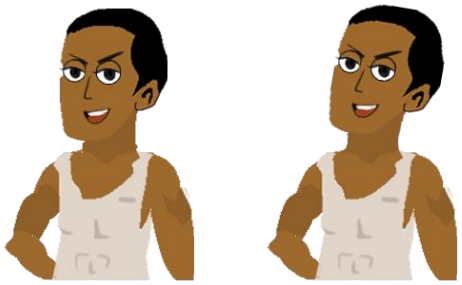
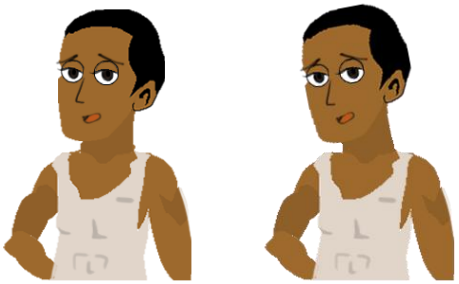
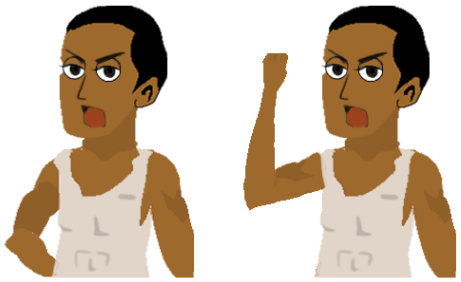
Figure 30: Rick's expression chart

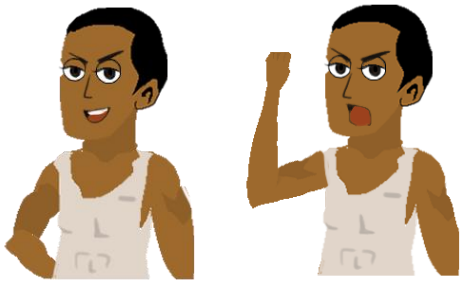
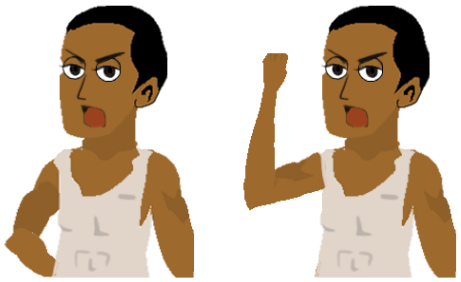
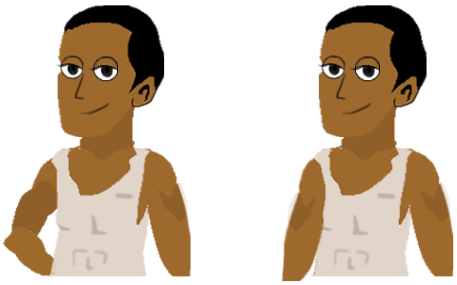
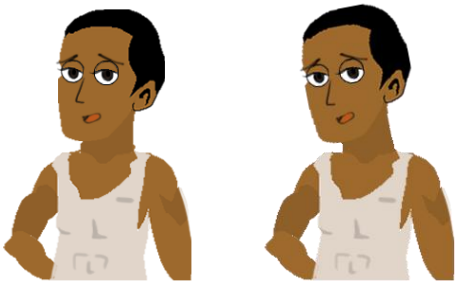
Figure 30 illustrates Rick's expressions and personality. Rick wears a simple jersey and it constantly gets wet because of the sweat from continuous walking. Although his gestures can be intimidating sometimes, Rick has a compassionate heart and looks out for the users all the time.

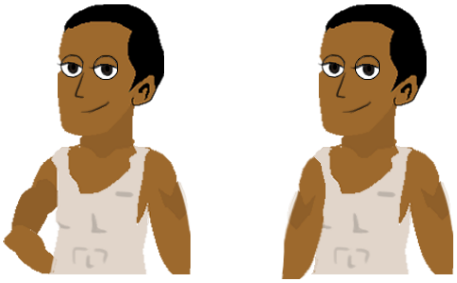
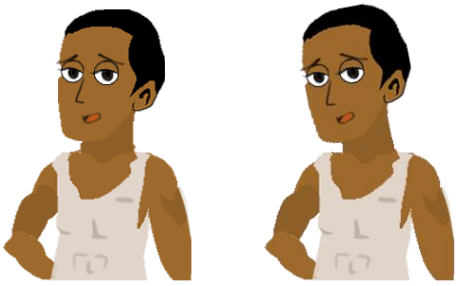
Table 3 below is the complete dialogue design for Rick.

Table 3: Rick's Lines

Condition	General Message	
Application start	Hey, I'm Rick. Look outside, isn't this a wonderful time for walking?	
First meeting	Greetings. I am coach Rick and I will be your coach. You are expected to walk with me every day, and soon you will have the body type you've been longing for.	
No walking in 1 day	Walking should be a daily activity! It's good for you! Let's go!	

Non-stop walking for 4 hours	Props for that. You are indeed a hardcore walker. What now, take a break or keep going?	
Walking speed too fast (in a car, biking, running, etc.)	Either you became a cheetah or you are in some kind of vehicle... I don't think you are a cheetah.	
Mission 80% complete	You are so close to completing your goal! Can't stop now! Come on! Let's go!!	
Mission complete	You've reached your goal!! I can't wait! What's your next mission??	

Territory captured	This is not enough yet. Look, we've got time to capture more territories!	
Territory ownership lost	Unacceptable! You've got to own the street, man!	
Territory captured(allian ce)	Soon we will take over the neighborhood. You are gonna help the rest get it. Right?	
Territory ownership lost(alliance)	That sucks, man. You've got to keep up your game.	

New member joined alliance	Someone joined our group. Nice.	
Member left alliance	Someone left. Well, I know that this can be exhausting.	

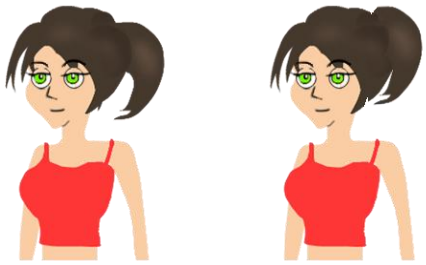
4.5.3 Sabrina

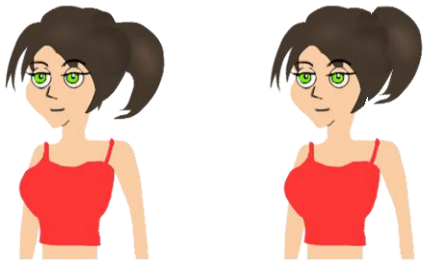
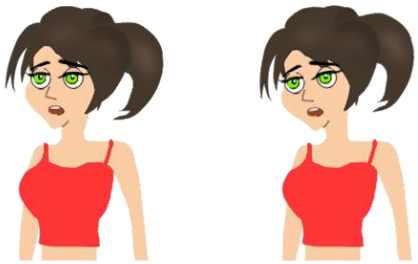
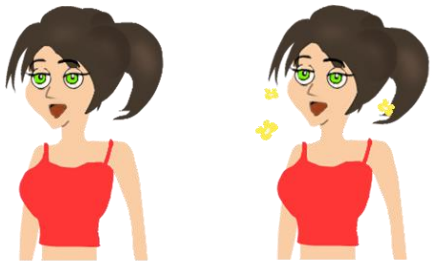
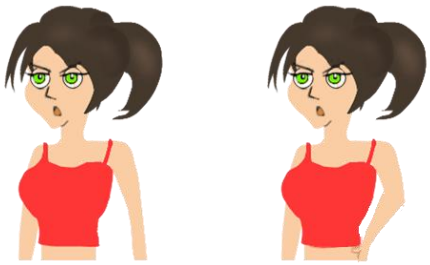
Sabrina is a friendly coach. She hardly gives hard trainings because she believes the most important thing is having fun while walking. Although Sabrina is not quite mature due to her young age, she is very easygoing, compassionate and cheerful. Figure 31 below is an expression chart of Sabrina.

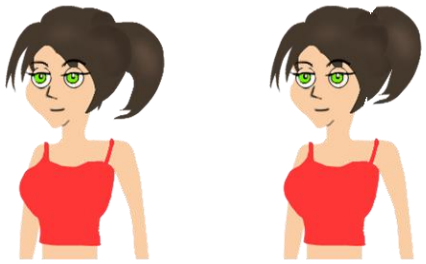
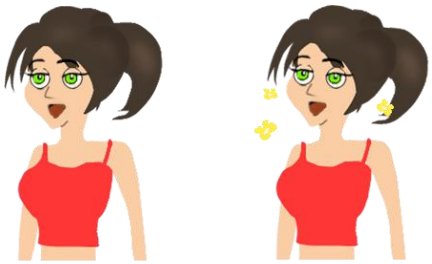
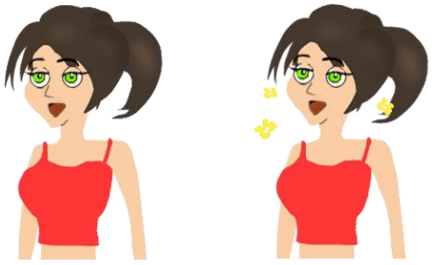
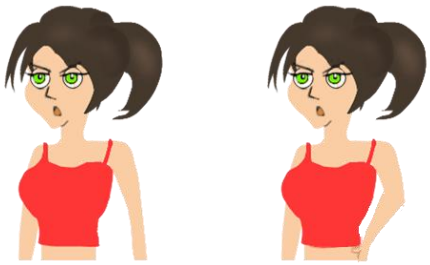
Figure 31: Sabrina's expression chart

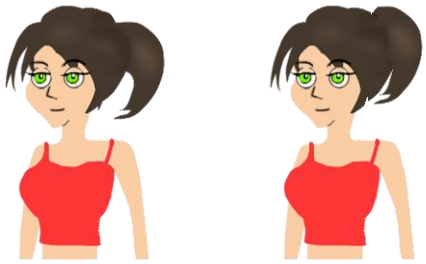
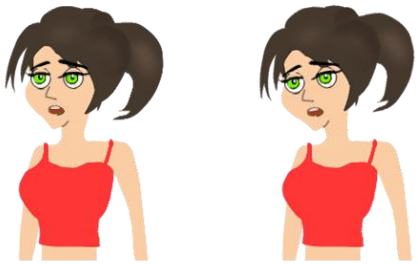
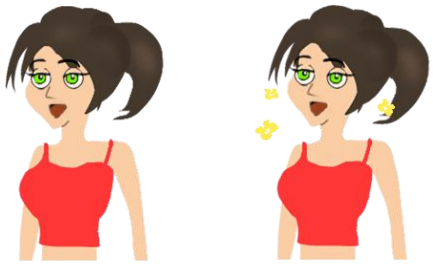
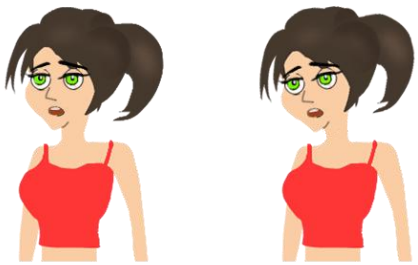
Figure 31 illustrates Sabrina's expressions and personality. As a friend, Sabrina wears casual clothes. She likes to wear her tank top because it gives her flexibility during work out. She has her pony tail all the time for the same purpose. Sabrina gives friendly and constructive advice. Table 4 below is the complete dialogue design for Sabrina.

Table 4: Sabrina's Lines

Condition	General Message	
Application start	Hi there, I'm your coach Sabrina.	

First meeting	Nice to meet you. I am a fitness trainer and my name is Sabrina. Let's take a walk!	
No walking in 6 days	I haven't seen you for a while…	
Non-stop walking for 40 minutes	Wow! Amazing stamina you've got there!	
Walking speed too fast (in a car, biking, running, etc.)	Cheating is NOT okay.	

Mission 80% complete	Look! We are almost there! Keep up!	
Mission complete	Congrats!! I knew you could do it!	
Territory captured	Awesome! You own this street!	
Territory ownership lost	Oops...	

Territory captured(alliance)	What a powerful alliance we have!	
Territory ownership lost(alliance)	Hey, don't be sad. We can get it back!	
New member joined alliance	Welcome! Glad to see a pair of new feet!	
Member left alliance	I'm sure he left by mistake.	

4.5.4 Samantha

Samantha is a cheerleader coach. Samantha has her pom-poms most of the time. She is extremely compassionate and cheerful. However, she is not mature and will drop the pom-

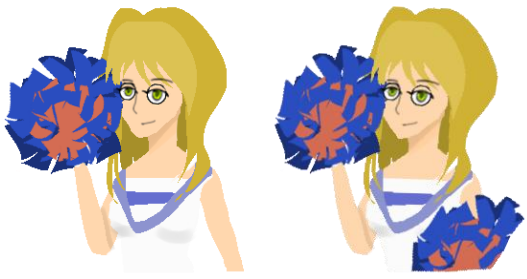
poms when she's frustrated. Samantha is not an aggressive coach, but she enjoys helping the user do well by giving the user hard training. Figure 32 below is an expression chart of Samantha

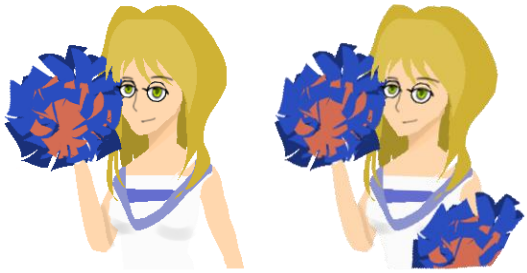
Figure 32: Samantha's expression chart

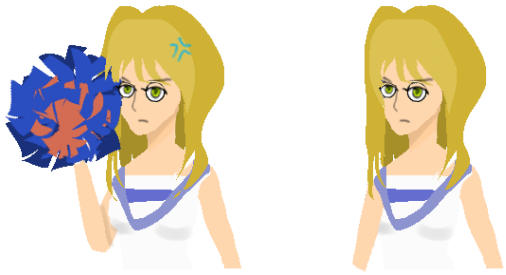
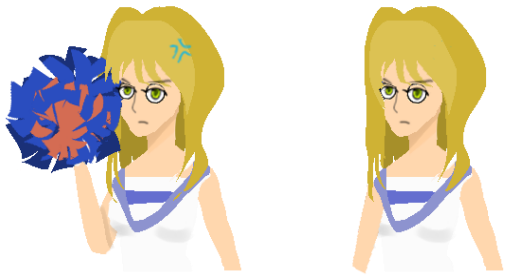
Figure 33 illustrates Samantha's expressions and personality. Samantha wears a cheerleader uniform. Her pom-poms match the color of her outfit. She is a bubbly blonde who enjoys cheering people up in her uniform.

Table 5 below is the complete dialogue design for Samantha.

Table 5: Samantha's Line

Condition	Lines	
Application start	Hi, I'm your cheerleader, Samantha.	

First meeting	Hi, my name is Samantha. I'm happy to be your cheerleader!	
No walking in 4 days	Hey... where have you been?	
Non-stop walking for 1 hour	Great! You should take a break though.	
Walking speed too fast (in a car, biking, running, etc.)	You are not superman yet...	

Mission 80% complete	Go! Go! Go!! (Or, Go [username] Go!! if possible)	
Mission complete	Yayyy!! You did it!	
Territory captured	This street is ours!	
Territory ownership lost	Awww... Are you gonna get it back?	

Territory captured(alliance)	Our alliance is the best!	
Territory ownership lost(alliance)	I'm sure we can get it back!	
New member joined alliance	WELCOME to our alliance!	
Member left alliance	Someone left our alliance... but it's ok! Cheer up!	

4.5.5 Steve

Steve is a nerdy coach. Some say he is just laid-back, some say he probably is not enthusiastic about walking at all. Steve is surprisingly strict and he is stingy with compliments. It has a lot to

do with his young age. On the bright side, he is unlikely to get aggressive. Figure 33 below is an expression chart of Steve

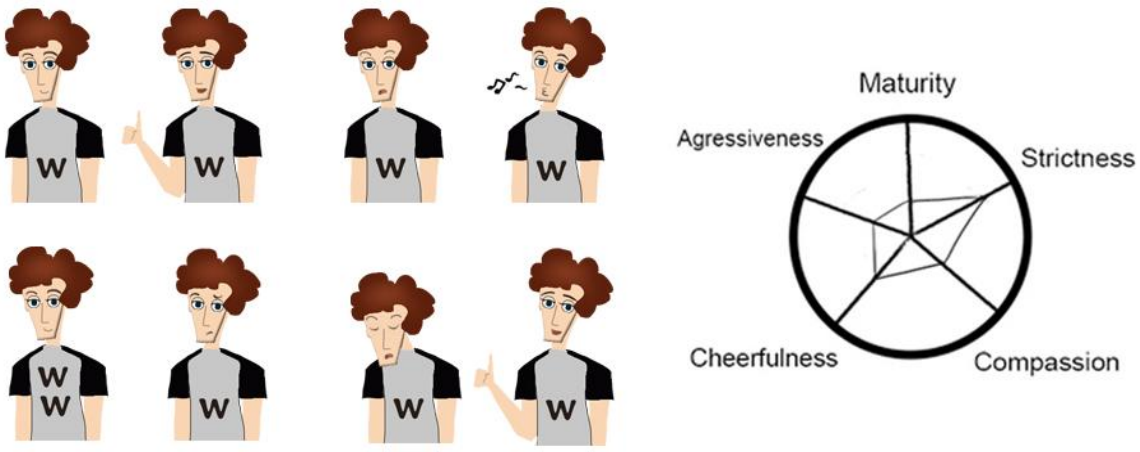
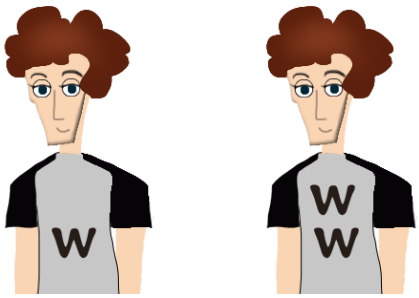


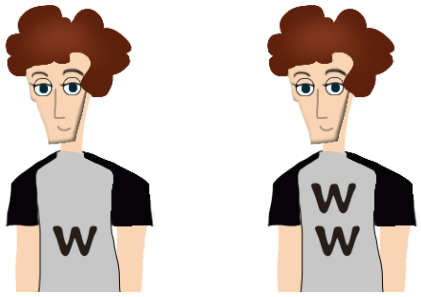
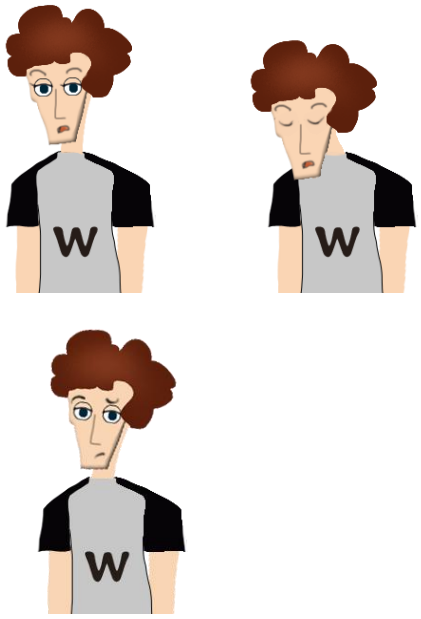
Figure 33: Steve's expression chart

Figure 34 illustrates Steve's expressions and personality. The nerdy coach Steve is always spotted wearing a baggy T shirt. However, the letter is W for Walking (and WW for We Walk) instead of C for Computer. Even though sometimes it's not that convincing to see him being a coach, Steve has his own strict program to make his trainee believe that he is serious about walking.

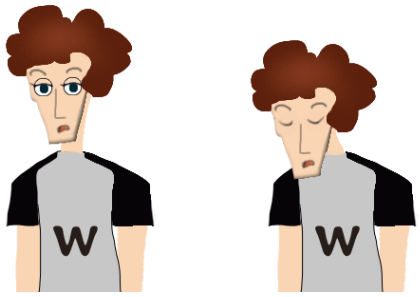
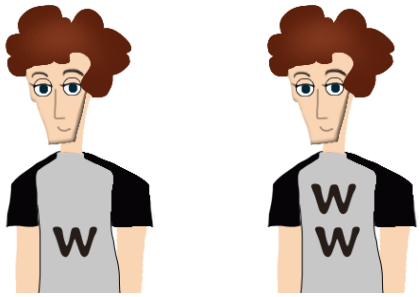
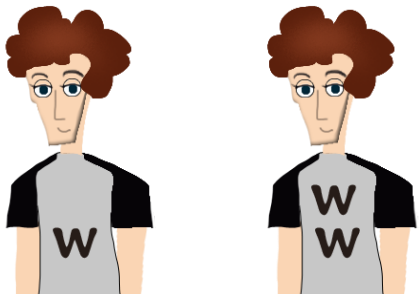
Table 6 below is the complete dialogue design for Steve.

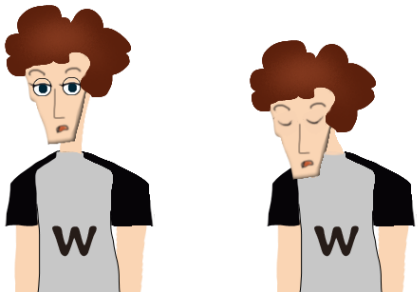
Table 6: Steve's Lines

Condition	General Message	
Application start	Hey, I'm your coach Steve.	

First meeting	My name is Steve. Glad to be your coach.	
No walking in 10 days	Long time no see, pal.	
Non-stop walking for 2 hours	Awesome. I've got to give you credit for that.	

Walking speed too fast (in a car, biking, running, etc.)	Dude, stop cheating. I can tell.	
Mission 80% complete	Keep going. Your goal can be reached.	
Mission complete	Very well done.	
Territory captured	Good job. Let's get some more.	

Territory ownership lost	Huh? What happened?	
Territory captured(alliance)	Our alliance captured a street.	
Territory ownership lost(alliance)	It's cool. Not our fault.	
New member joined alliance	Hey! A newbie just walked in.	

Member left alliance	Do you remember that guy? Yeah, I think he just left.	

4.5.6 Tom

Tom is a businessman coach. He is the users' co-worker...who can give promotions. Tom is quite mature and strict. His philosophy is: walking gets one a promotion and slacking gets one fired! Tom is aggressive when a user has not been active for days. He cheers when the user achieves his or her goal, but his compassion runs a little bit low. Figure 34 below is an expression chart of Tom

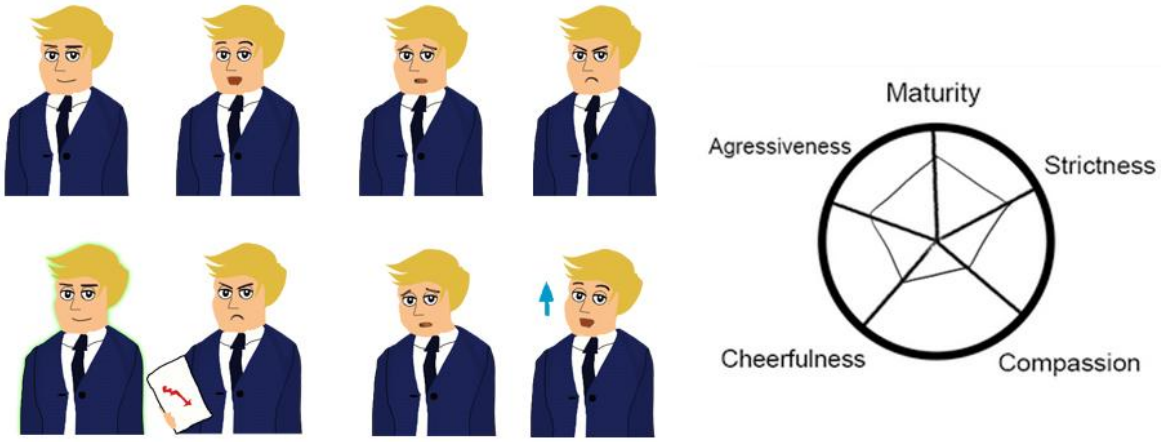
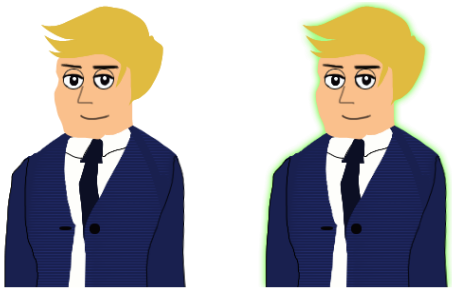
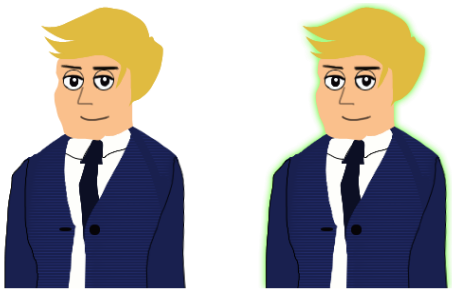
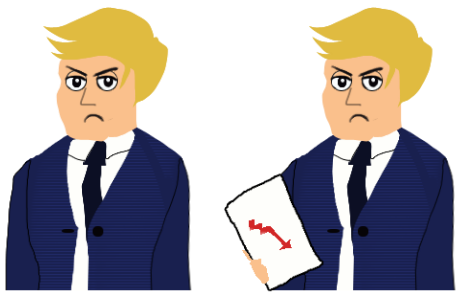


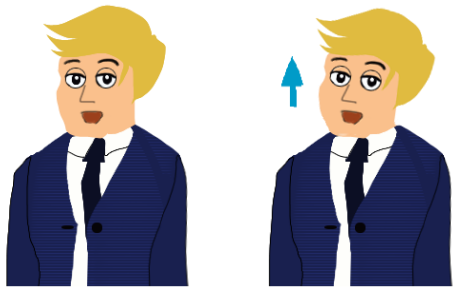
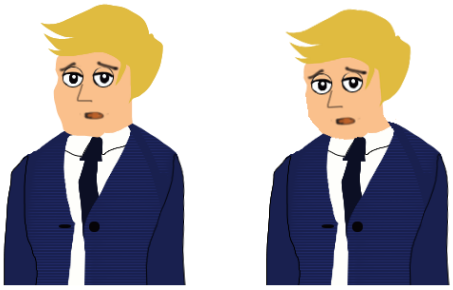
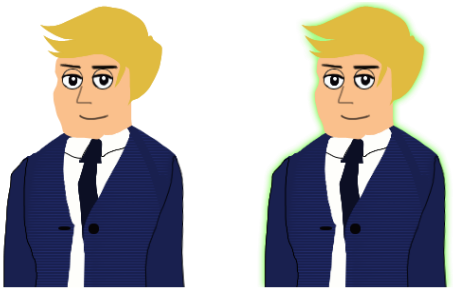
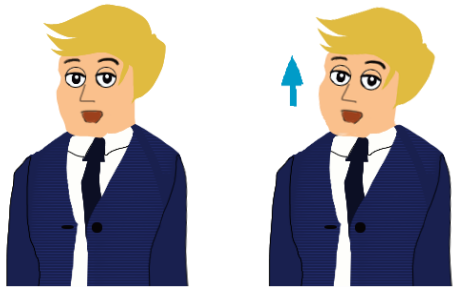
Figure 34: Tom's expression chart

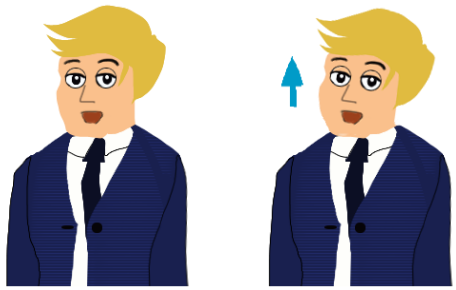
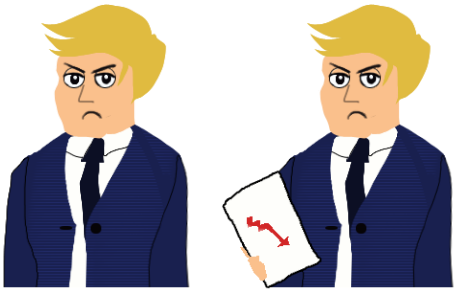
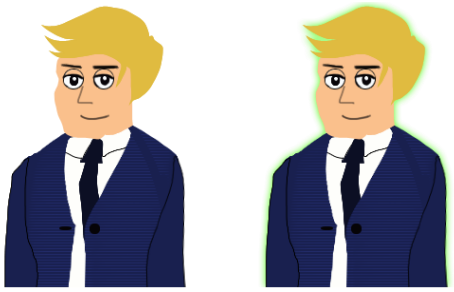
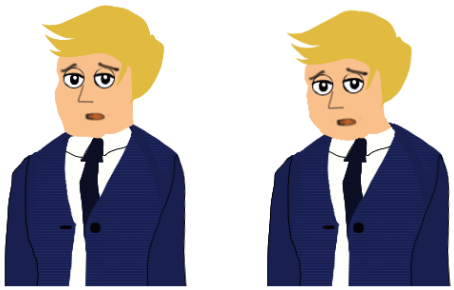
Figure 34 illustrates Tom's expressions and personality. As a dedicated businessman, Tom wears his blue suit and tie every day. He works at a Top 500 company, but it is no secret that he sees exercise as important as career. Tom does not have time to go on a hiking trip, but he does prefer walking to driving when he goes to work every morning. Tom is a great coach because he knows that a fit, healthy body comes before any career.

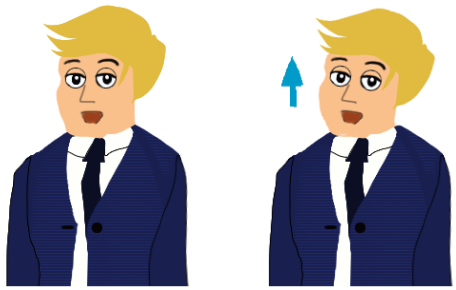
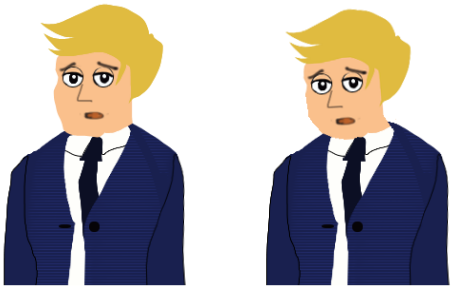
Table 7 below is the complete dialogue design for Tom.

Table 7: Tom's Lines

Condition	General Message	
Application start	Hello. I'm your coach, Tom. Nice to see you today.	
First meeting	Hi. My name is Tom. I am a businessman and I devote my spare time training folks like you.	
No walking in 3 days	Slacking off, are we?	

Non-stop walking for 1 hour	Great! You've been walking for a really long time!	
Walking speed too fast (in a car, biking, running, etc.)	We can't count this. This isn't walking.	
Mission 80% complete	Guess what? You are only a little bit away from reaching your goal!	
Mission complete	End of mission! You get a promotion!	

Territory captured	This street is under your name.	
Territory ownership lost	We should get back the ownership. I might lose my job because of this!	
Territory captured(alliance)	Our company is doing well. You get a raise.	
Territory ownership lost(alliance)	Stop counting on that bonus. We lost the ownership.	

New member joined alliance	Hey, we have a new member joining us!	
Member left alliance	Someone just got fired in our alliance.	

4.6 Eastern Coaches

The Eastern style is heavily influenced by the Japanese anime style, which is very exaggerated and adds lots of expressiveness to the pictures. In order to appeal to various audiences' tastes, the Eastern coaches offer a wide range of choice, ranging from school girls to robots and lion man, as shown in Figure 35:

Figure 35: Group shot of Eastern coaches

Details on individual coaches are discussed in the following sections.

4.6.1 Aldin

Aldin is a mysterious lion man who came from the forest. No one knows much about his past or the reasons why he joined the *World Wide Walk* coach team, but everyone certainly feels the power and dignity of the king of the wild through his existence. Rumors have even suggested that he actually has a crown. As a coach, Aldin is extremely strict and aggressive. For those who seriously have the heart to conquer the world, Aldin is the best choice. Figure 36 shows the initial concept sketch of Aldin.

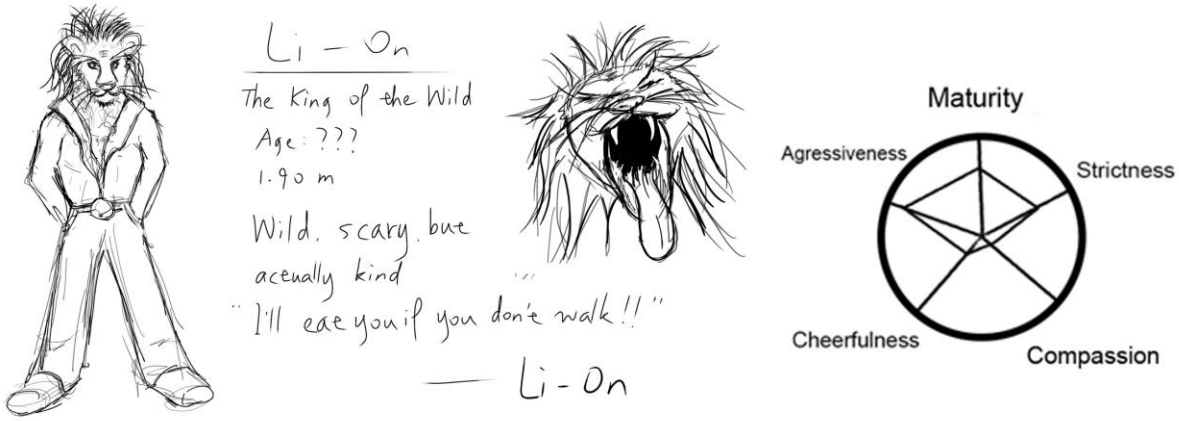


Figure 36: Initial concept sketch of Aldin

As shown in Figure 36 above, in the initial design Aldin was named Li-On, which was simply a variation of the word lion. Some characters with that name had already been designed in other games and eventually the idea was discarded. Aldin sounds like the name of a mythical creature and it seemed to fit this lion man character.

Figure 37 below shows the development of Aldin. The shadows were enhanced on his face to give more definitions to his facial features. Long whiskers were also added which is what a big cat like Aldin should have.

Version.1 Final Version

Figure 37: Development of Aldin

Aldin is wild and furious. He gets irritated very frequently and easily. However, Aldin is more than a yelling beast because sometimes the anger is his way to get the user motivated. He possesses the heart of a lion and conquering the world in *World Wide Walk* is truly his only focus. Aldin has many furious expressions where his claws and fang are revealed, as shown in Figure 38 and Table 8.

Figure 38: Aldin's expression chart

Table 8: Aldin's Lines

Condition	Lines	
Application start	Look at the crown. I am the king of the wild and together we will conquer the world. Now start today's walking. Hello. Today is a good day for hunting… I mean walking.	 

First meeting	Greetings. I am Aldin, your coach for World Wide Walk. Our goal is simple and clear: Conquer the world. Are you prepared?	
No walking in 12 hours	You have to fight just to survive in the wild! What do you think you're doing?	
Non-stop walking for 2 hours	Not too bad. Keep moving.	
Walking speed too fast (in a car, biking, running, etc.)	Don't play tricks with me! It's not gonna work!	

Mission 80% complete	Still not there yet. Keep walking!	
Mission complete	You finally did it. I'll give you some rewards for the effort.	
Territory captured	Not bad at all! Keep it up and get more! Well done! But remember this is only one tiny step toward conquering the world!	 
Territory ownership lost	Hmm…They dare to take our street? Time to think about fighting back. Time to get it back! Show them what you're made of!	 

Territory captured(alliance)	Hahaha. Our team did a good job. We will conquer the world very soon!	
Territory ownership lost(alliance)	How dare they challenge us! We'll fight back strong! Now go out already and take their territories!	
New member joined alliance	Hmm… Let's see what this guy is made of. Hopefully they can be of some use.	
Member left alliance	Let it be. One tiny loss wouldn't slow our step.	

4.6.2 Catherine

Catherine is a cheerful college sports girl. The way she dresses up closely resembles women tennis players. She is passionate about sports and she feels really excited to join the *World Wide Walk* program. Catherine is not very aggressive and strict. She likes to enjoy the pure joy of walking without having to worry about capturing territories and competing with other users. Figure 39 shows the initial concept sketch of Catherine.

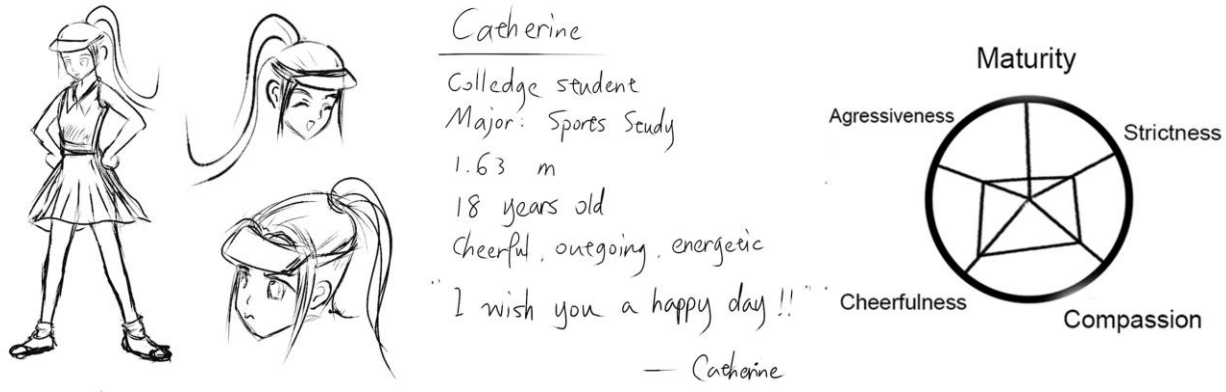


Figure 39: Initial concept sketch of Catherine

Catherine was the first coach designed in the Eastern style. Figure 40 below shows the development of Catherine. As soon as the finished image (version.1) was seen on the phone screen, the team realized that she needed a lot more contrast and thicker edge lines. Thus the contrast of the image was increased and the lines were thickened. The top of Catherine's ponytail was fixed so that it would not look like it was cut off.

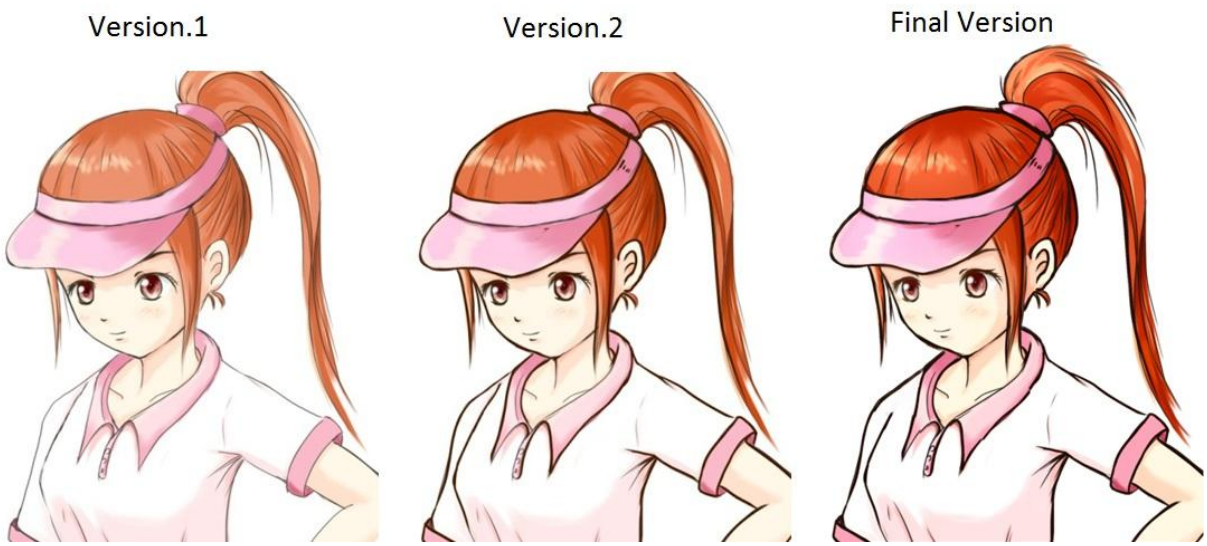


Figure 40: Development of Catherine

Catherine is a very expressive character, as shown in Figure 41 and Table 9 below. She wears a big smile on her face most of the time and speaks encouraging words to the user. However, she turns angry really fast if laziness is spotted.

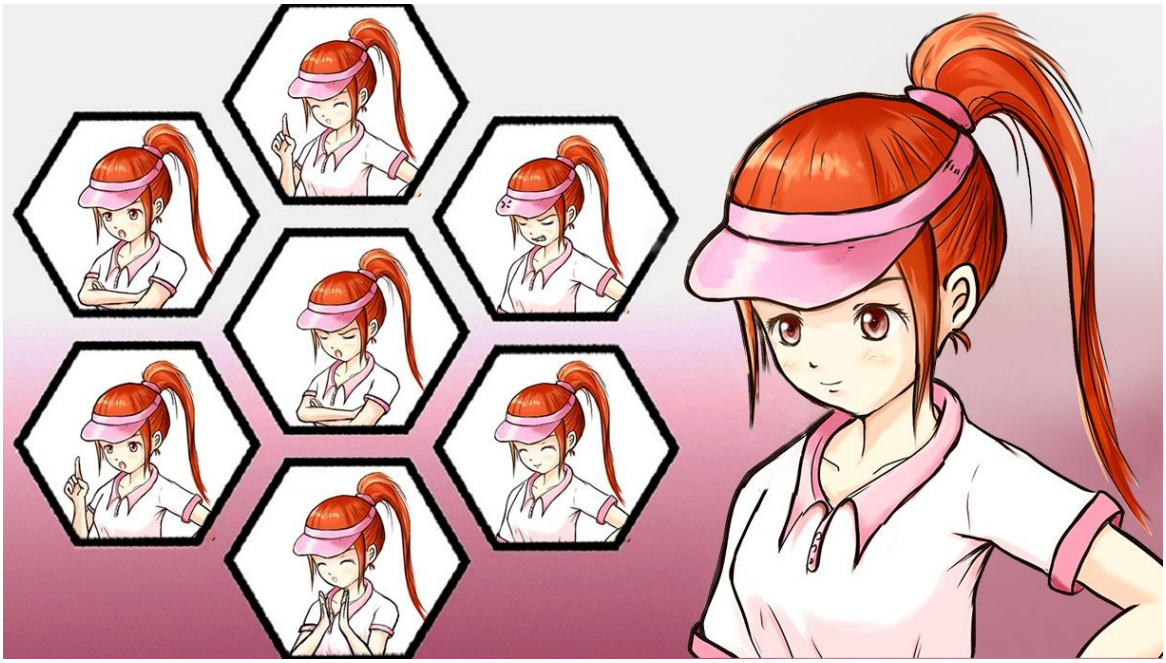


Figure 41: Catherine's expression chart

Table 9: Catherine's Lines

Condition	Lines	
Application start	Hi! Ready for today's walk?	
First meeting	Hello I am Catherine your new coach and friend! No worries! Under my guidance, you will become a great walker very soon!	 
No walking in 3 days	You cannot be like this! Come on get off your couch! How much more time do you want to waste on your couch? Do you want to become a potato?	 

Non-stop walking for 1 hour	Good job! Keep it up and let's gather even more points!	
Walking speed too fast (in a car, biking, running, etc.)	Hey you are not following our rules are you? I want to remind you that speeding will not get you any points!	 
Mission 80% complete	We're almost there! Go go go!	
Mission complete	Yay we did it! Congratulations! You completed the mission!	

Territory captured	Wow! Maybe we can really capture the world one day!	
Territory ownership lost	Hey! I don't want you to lose to them! We need to work harder. Come on!	
Territory captured(alliance)	Wow! Another addition to our territory!	
Territory ownership lost(alliance)	Ahh what a loss for the team. Let's work harder and get it back!	

New member joined alliance	We are becoming a big family! (XXX joined your alliance)	
Member left alliance	What are they thinking? They will regret this one day! (XXX left your alliance)	

4.6.3 Jack

Jack is the most energetic of all the coaches. Although Jack is already 19 years old, the playfulness of a little boy can still be found on him. However, when it comes to walking, Jack is as serious as any other professional coaches. He will ensure that the user put his/her best efforts to become a winner of this game. Figure 42 shows the initial concept sketch of Jack.

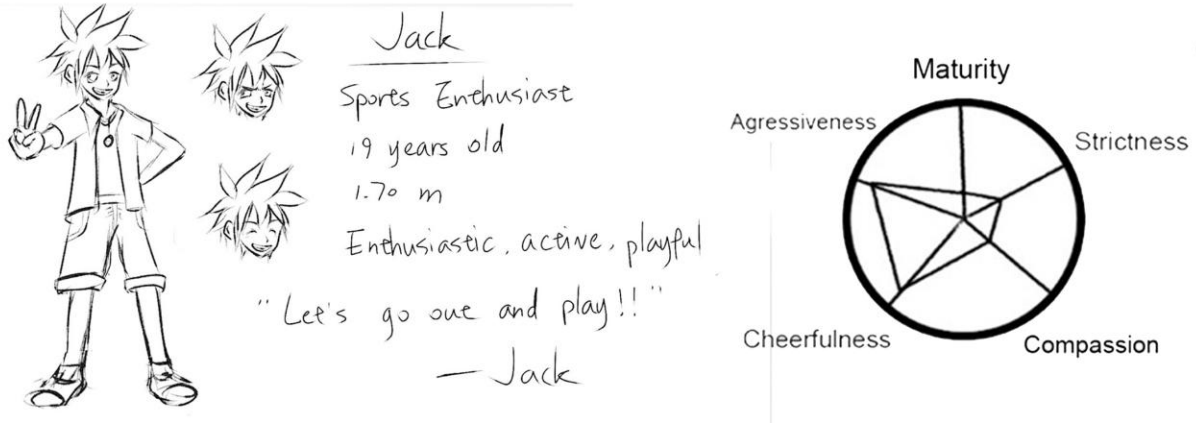


Figure 42: Initial concept sketch of Jack

Figure 43 below shows the development of Jack. His reversed hand in the final version was fixed and color was added to his watch. His eyebrows were made thicker and his face a little slimmer to give him a more handsome look. His eyes are changed into smiling ones because this expression is supposed to be very happy and cheerful!

Figure 43: Development of Jack

Jack has an energetic and cheerful look for most of the time due to his personality. However, users will find him scratching his head if they keep being lazy or when unpleasant things happen. Jack is really competitive and loves to play games. His eyes will be burning with excitement when challenges appear. Details of all his various expressions can be found in Figure 44 and Table 10 below.

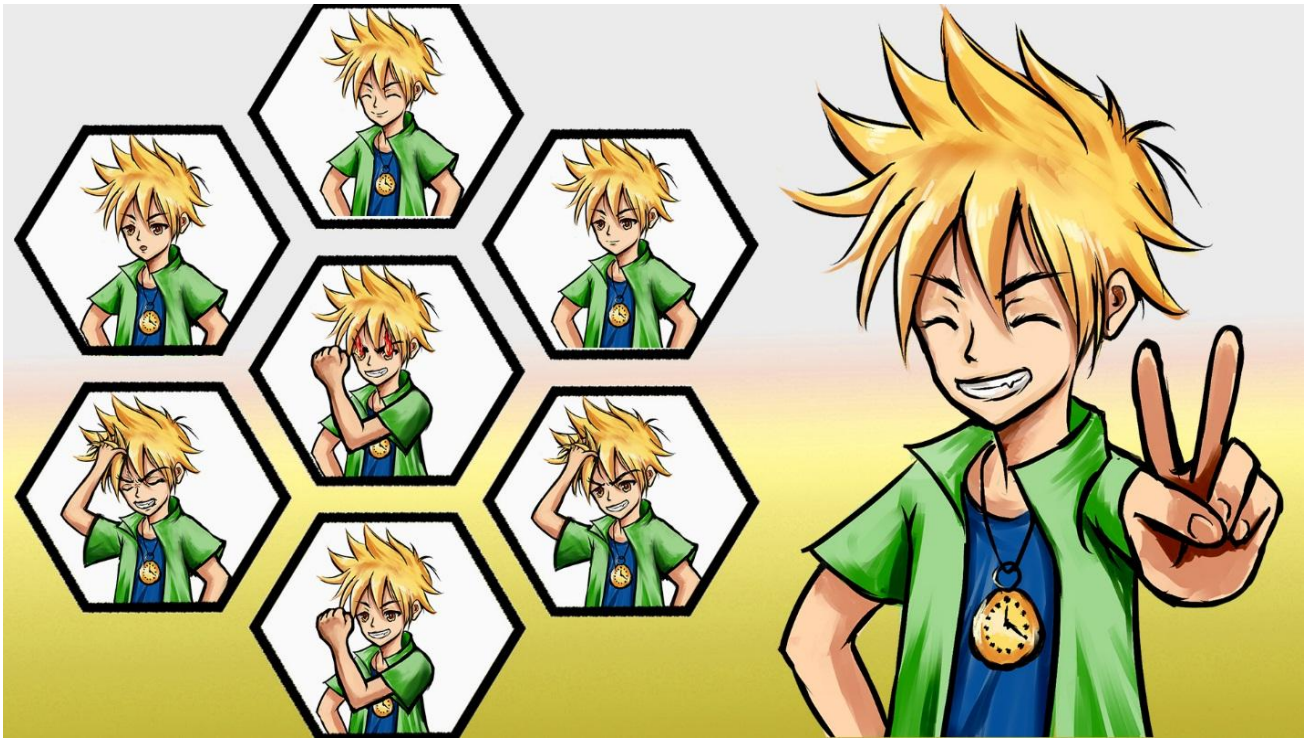


Figure 44: Jack's expression chart

Table 10: Jack's Lines

Condition	Lines	
Application start	Yo! My legs are anticipating. Let's go! I cannot wait any more! Time to walk!	 

First meeting	Nice to meet you! I'm Jack. Together we will be the best combination in this game! You'd better be prepared for the coming hard work.	
No walking in 2 days	Err you're giving me headaches… What can we achieve by staying home every day? Err…This is not good. You've got to get walking now.	 
Non-stop walking for 1 hour	Nice effort! But it's only the beginning of today's exercise!	
Walking speed too fast (in a car, biking, running, etc.)	Hey hey trying to cheat? If there is an easier way to do this I would have told you already!	

Mission 80% complete	80% already! The reward is waiting ahead!	
Mission complete	Haha we did it! I know you can do it when you have the heart.	
Territory captured	Excellent! We are gradually capturing this area! Aww I'm excited! We're going to be the best!	 
Territory ownership lost	Ahh how could this happen? Our opponents are stronger than expected.	

Territory captured(alliance)	We will conquer the world!	
Territory ownership lost(alliance)	Oh no our street! Darn it! These guys do have some guts! Time to fight back!	 
New member joined alliance	I'm excited! With such an expanding team, being top of the world is no longer a dream! (XXX joined your alliance)	
Member left alliance	Ahh I hate this kind of thing. Headache… (XXX left your alliance)	

4.6.4 Rebecca

Rebecca is a quiet and elegant girl. She's the most compassionate one among all the coaches. She kindly encourages the user for their achievements while patiently repremending them if any trait of laziness is spotted, almost like a gentle elder sister. Also, Rebecca is a true master of Yoga and she has a strong connection with nature. Figure 45 shows the iinitial concept sketch of Rebecca.

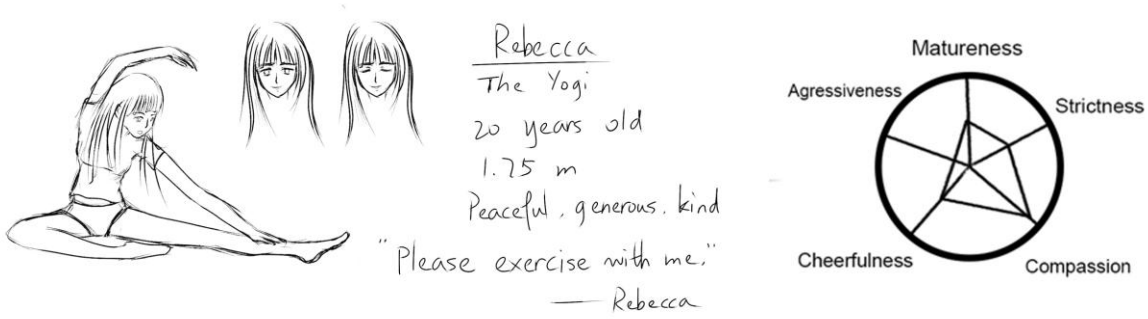


Figure 45: Initial concept sketch of Rebecca

When it comes to the actual design, Rebecca is the one with the most revisions of all the coaches. The illustration of Rebecca (Figure 46) was drawn at the very early stage of the project. Her proportions did not suit as a usable asset, but it was an attempt to capture the feeling of the character.

Figure 46: Rebecca early illustration

Rebecca retained a sense of peace and steadiness throughout her illustration phase, but she looked a little too young. In order to create a strong contrast with the other female coach, Catherine, Rebecca had to look more mature. As a result, the final design shown in Figure 47 was drawn.

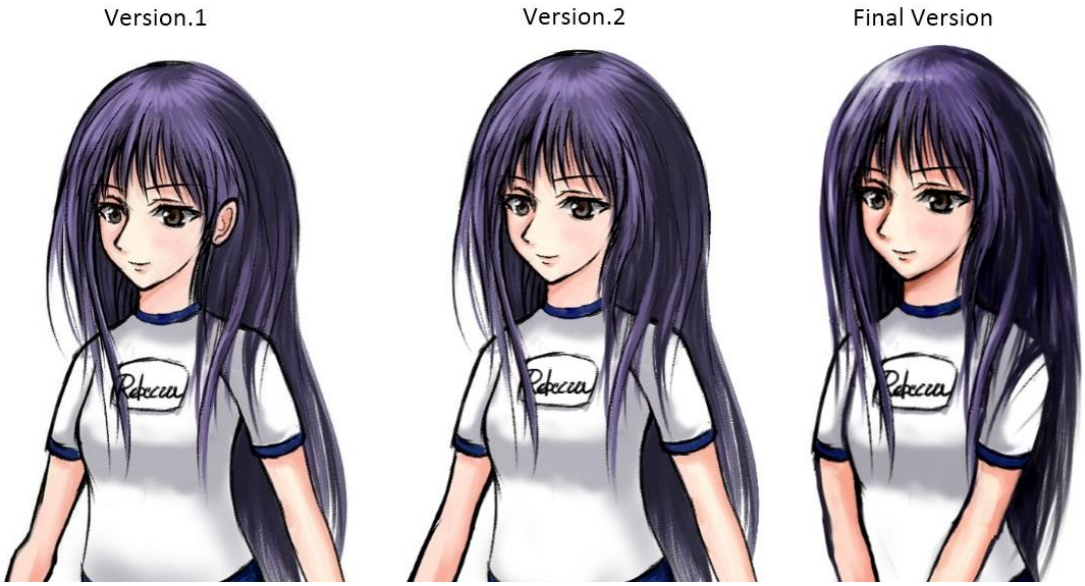


Figure 47: Development of Rebecca

Some viewers suggested that her ear appeared to be in a strange position in Version.1. So in the final version her hair now covers her ear. Some other changes were made such as the shadow on her belly was fixed, her hair was less messy, and added more contrast and highlights to the image.

Rebecca is not as expressive as the other coaches due to her peaceful nature, as shown in Figure 48 and Table 11 below. For most of the time, she keeps calm and alleviates users' worries. However, if the user slacks off too much, even Rebecca will get a little bit angry.

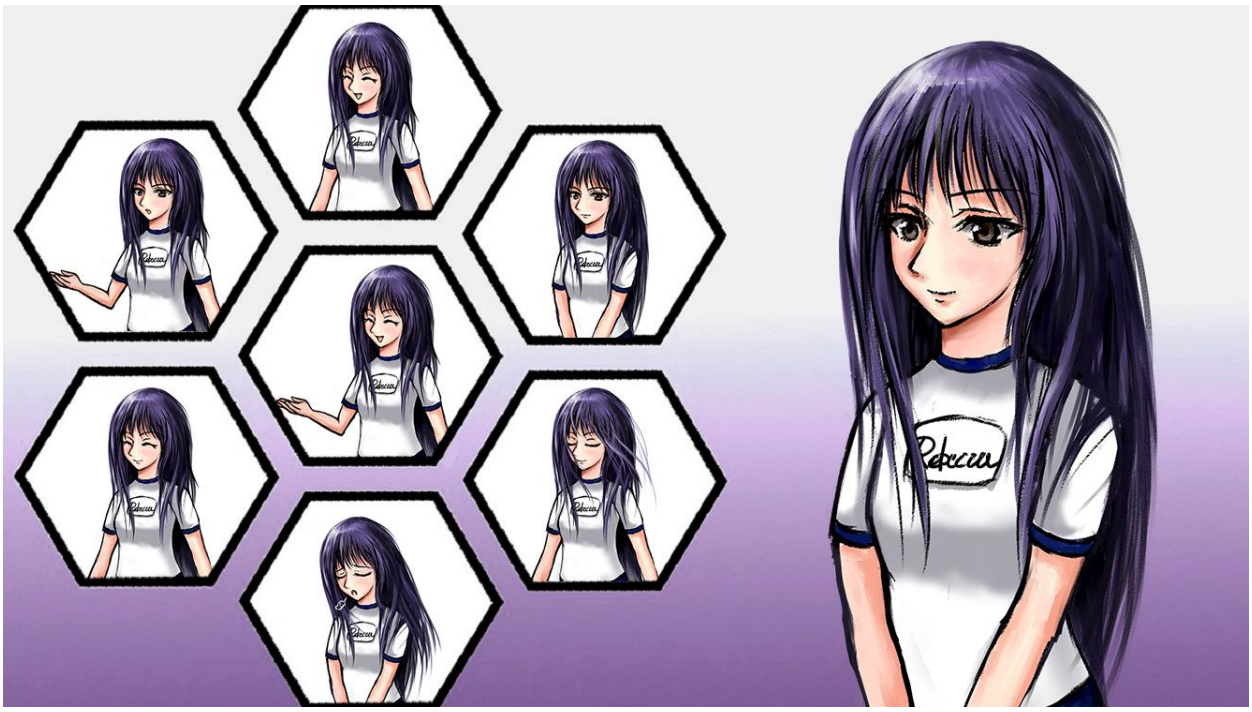


Figure 48: Rebecca's expression chart

Table 11: Rebecca's Lines

Condition	Lines	
Application start	Listen, the wind is calling us. Let's go. Good day! I'm looking forward to walking with you~	 
First meeting	Nice to meet you. I am Rebecca, your new coach for World Wide Walk. I will assist you in any way I can.	
No walking in 3 days	You're being too lazy. I will get angry if you keep doing this. Come on. There are many wonderful things outside. The wind, the sunlight and the trees are waiting for us.	 

Non-stop walking for 30 mins	I'm glad you're keeping it up~ Keep going!	
Walking speed too fast (in a car, biking, running, etc.)	You cannot play against the rules! It's unfair to the other players and you don't get exercise.	
Mission 80% complete	The mission is almost completed! But don't push yourself too hard. Make sure you take a break.	
Mission complete	Congratulations! Please take your rewards!	

Territory captured	Woohoo! You are a great walker! Impressive. I believe you will become a great walker one day.	 
Territory ownership lost	We lost the street (sigh)… We have to walk more.	
Territory captured(alliance)	Our teammates are doing a great job! We should walk more to show our support!	
Territory ownership lost(alliance)	The opponents are not making it easy for us. Don't be depressed. Sometimes we gain and sometimes we lose. Keep walking and you will get what you deserve.	 

New member joined alliance	We got a new friend! Let's walk together~ (XXX joined your alliance)	
Member left alliance	We cannot force them to stay with us all the time. Let nature take its course.	

4.6.5 Rob

Rob is a professional fitness coach who has worked in the industry for more than 6 years. Rob is severely strict and walking with him would not be easy. Therefore he is recommended for those who aim to walk seriously. Figure 49 shows the initial concept sketch of Rob.

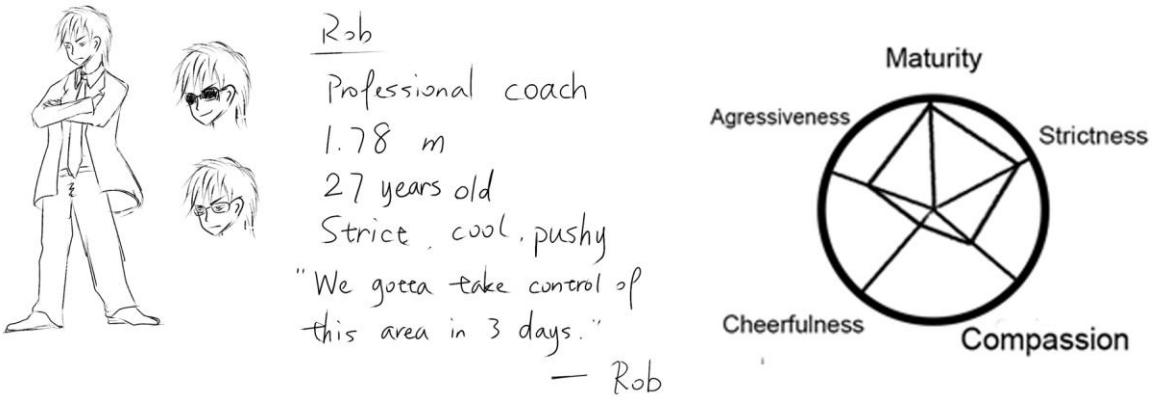


Figure 49: Initial concept sketch of Rob

When it comes to the final design, his hair was made shorter to give him a cleaner look. He did not look strict enough after Version.1 so his eyes were redrawn and some edge lines were enhanced to create the final image, as shown in Figure 50 below.

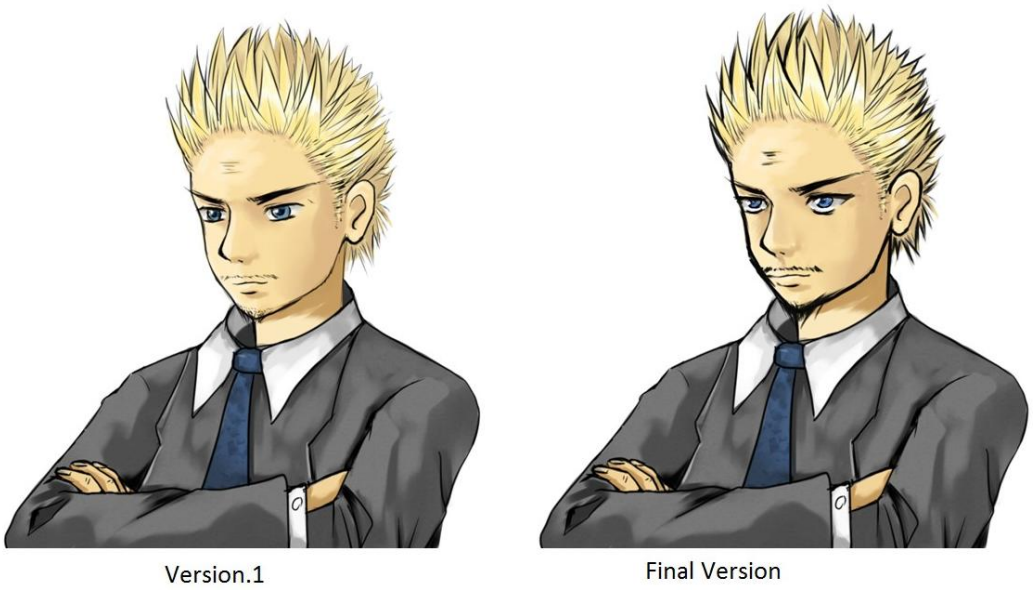


Figure 50: Development of Rob

This expression below (Figure 51) was eventually removed from his expressions because it looked sort of funny and did not suit Rob.

Figure 51: Rob's unused expression

The artist decided to make him always wear sunglasses when he is working (just a little funny fact to make him more likable). However, he is still the strict professional couch who frequently shouts at users' mistakes and laziness, as shown in Figure 52 and Table 12 below.

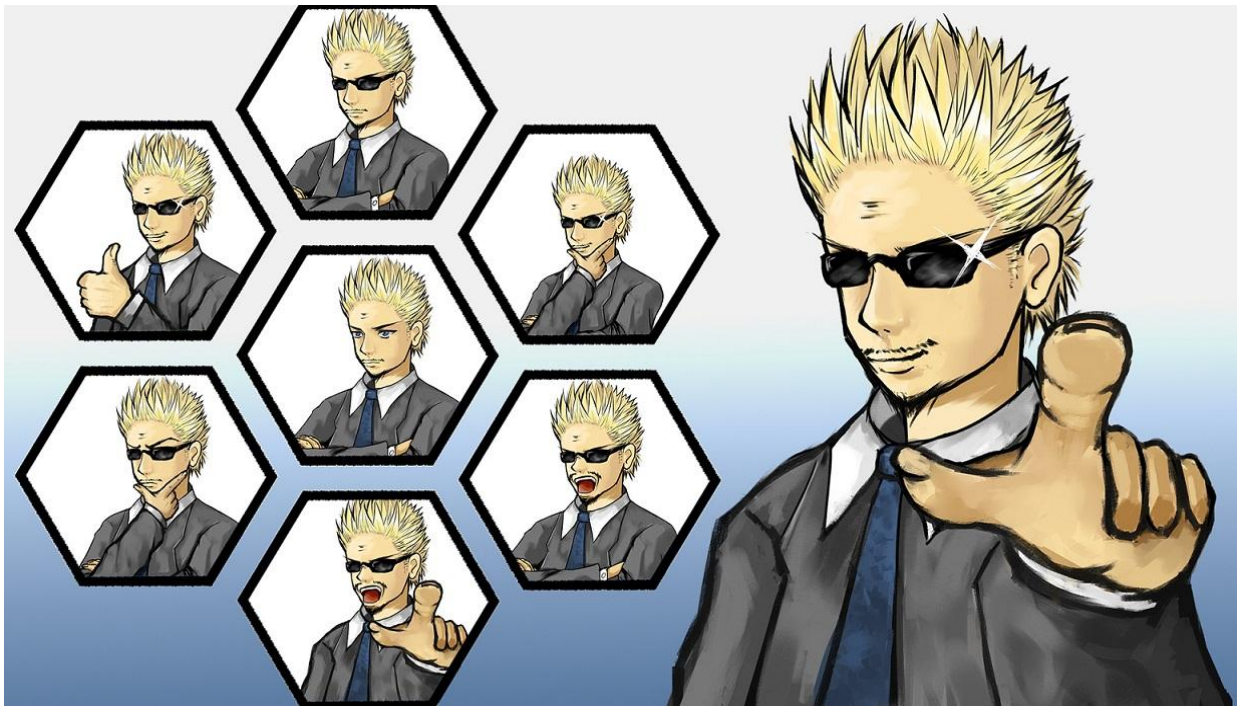


Figure 52: Rob's expression chart

Table 12: Rob's Lines

Condition	Lines	
Application start	What's up? Time to get to walk! (Time to put on his glasses actually...)	 
First meeting	Hello I am your new coach Rob. Let me get it straight, our one and only goal is to conquer the world. Get ready! By the way, could you please hand me that pair of sunglasses?	
No walking in 1 day	You just wasted 24 hours of our time! Are you kidding me? You've got to get it back today! Go! Right now!	 

Non-stop walking for 2 hour	That's what I'm talking about!	
Walking speed too fast (in a car, biking, running, etc.)	Don't try to be smart! This doesn't count!	
Mission 80% complete	Keep it up! Walk harder!	
Mission complete	Not bad!	

Territory captured	We've got to become the big landlord!	
Territory ownership lost	That's all you got? Are you kidding me?! Pull yourself together and get it back!	
Territory captured(alliance)	One step closer to conquering the world!	
Territory ownership lost(alliance)	They are still not giving up? Challenge accepted! Now we fight back! Go!	 

New member joined alliance	Nice! Now it's easier to gather points! (XXX joined your alliance)	
Member left alliance	Seriously? Those inconsistent people. Leave them alone. (XXX left your alliance)	

4.6.6 Toddo

Toddo is a little robot that is powered by users' walking energy. He has a refrigerator in his stomach, stuffed with ice-cold refreshing drinks. Personality-wise he averages all the five aspects on the star trait. However, if users want to keep a good friendship with Toddo, they'd better keep walking. Figure 53 shows the initial concept sketch of Toddo.

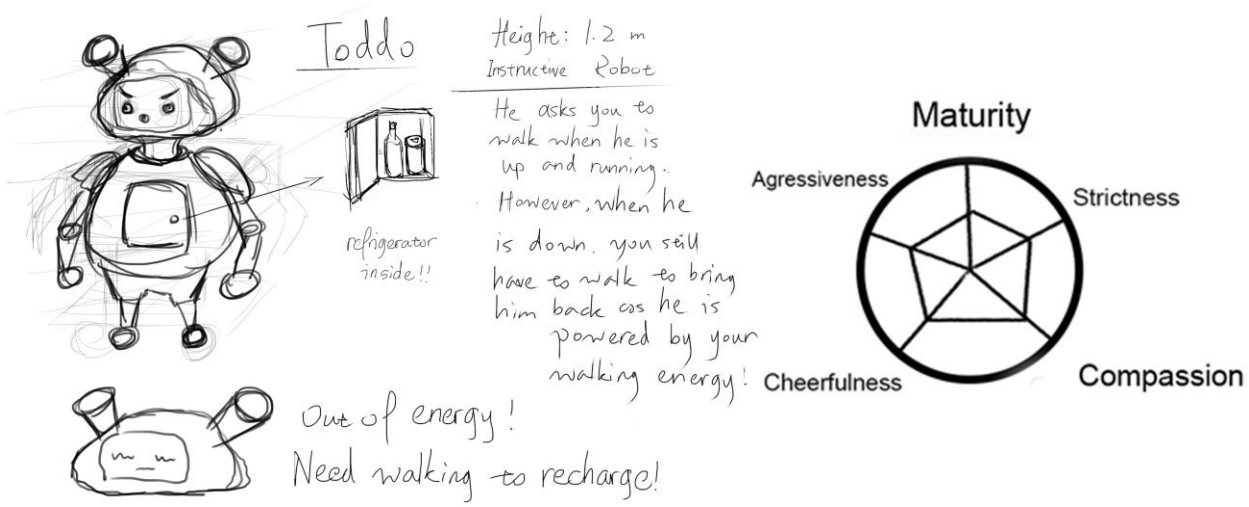


Figure 53: Initial concept sketch of Toddo

Initially the color scheme of Toddo was undecided so there are several variations (Figure 54).

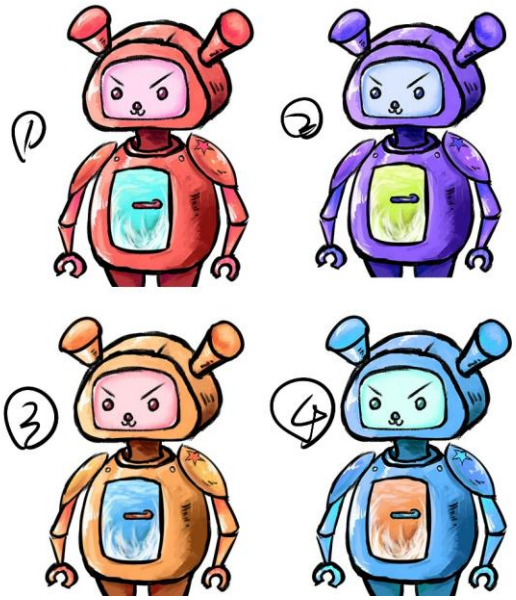


Figure 54: Toddo color variations

Eventually the orange variation was decided on (number 3) because it felt more energetic and fit well with the game.

Toddo's face is made of a 12" LCD screen that shows different expressions, as shown in Figure 55 and Table 13 below. As the newest generation *World Wide Walk* robot, he is intelligent enough to simulate several human emotions. His LCD face turns green when he is sick (out of power) and turns red when he is angry. His ears come down when he is frustrated. Although sometimes he still speaks like a robot, users will find themselves connected with Toddo just like with any other lively friends.

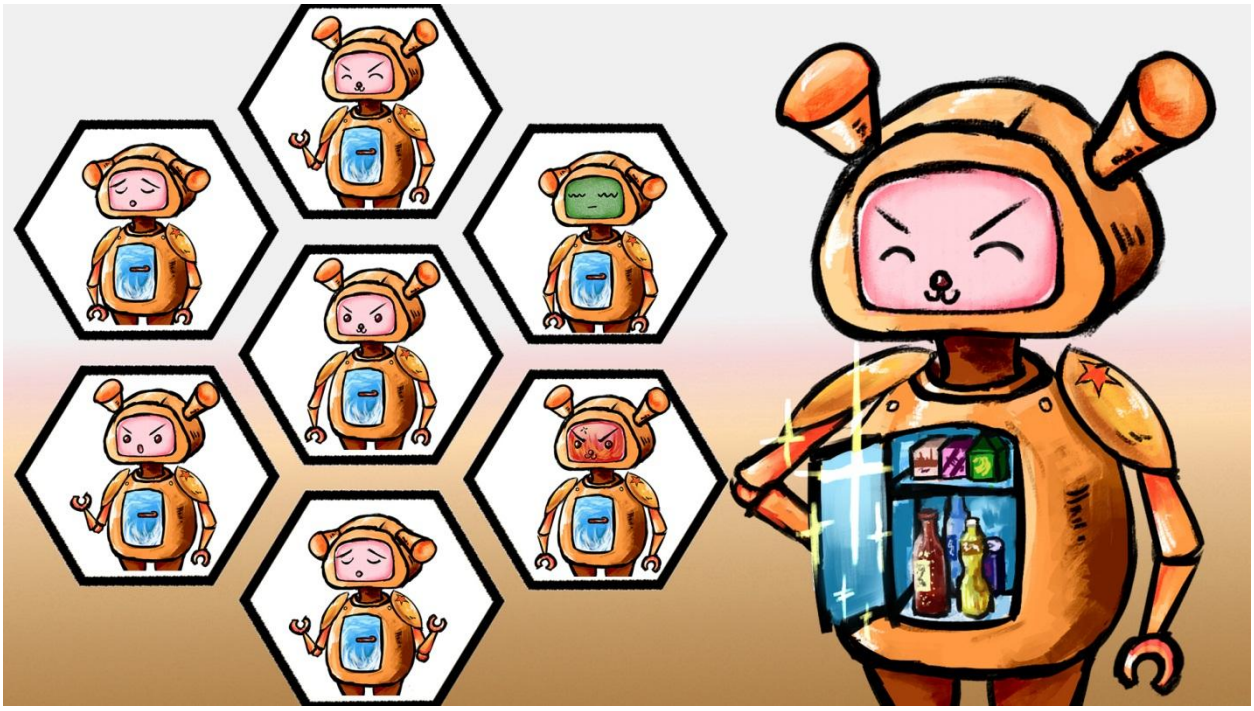
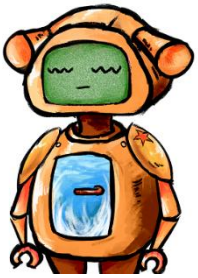
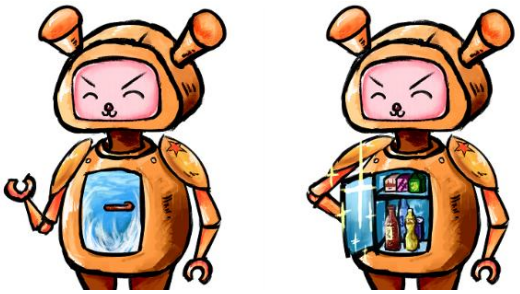


Figure 55: Toddo's expression chart

Table 13: Toddo's Lines

Condition	Lines	
Application start	Welcome to World Wide Walk. Time to start walking today.	
First meeting	Greetings! I'm the newest generation World Wide Walk robot Toddo. Nice to meet you.	 
No walking in 3 days	(Poor Toddo is out of energy right now. Please walk to bring him back.)	
Non-stop walking for 1 hour	Good job! How about a brief rest and take some drinks?	

Walking speed too fast (in a car, biking, running, etc.)	This is not allowed by the system. Please walk seriously.	
Mission 80% complete	You are almost there! Keep moving! Refreshments are waiting ahead.	
Mission complete	Congratulations! Take this as the reward!	
Territory captured	Well done! Territory expanded!	

Territory ownership lost	Territory lost. Please walk harder. How can we capture the town if you let them take a street?	
Territory captured(alliance)	As expected! Teamwork for the win! Now it's time for the party!	
Territory ownership lost(alliance)	They dare to take our territory! I am angry even though I'm a robot!	
New member joined alliance	Alliance size expanded! (XXX joined your alliance)	

Member left alliance	Different people walk different roads. It can't be helped. (XXX left your alliance)	

5 System Design

In order to create *World Wide Walk*, many smaller sub-systems were created to handle the various parts of the application. To understand the design of the system, it is easiest to look at how the whole system runs:

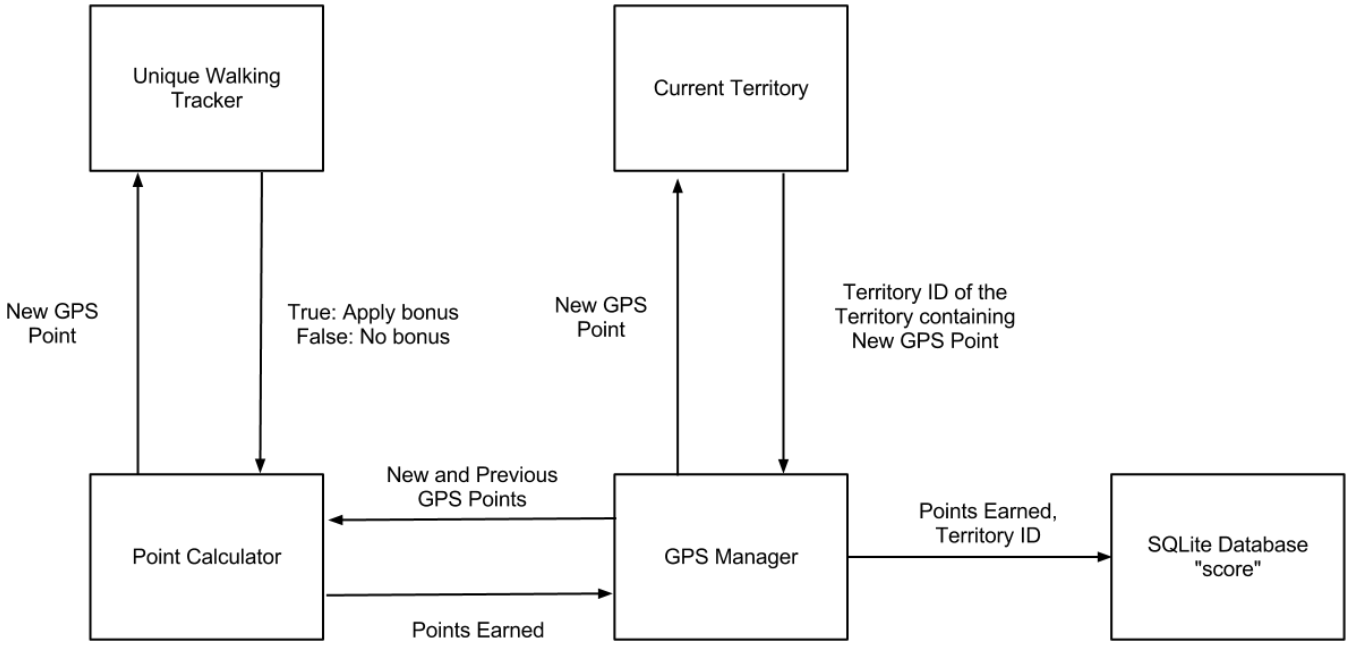


Figure 56: A diagram of the system design

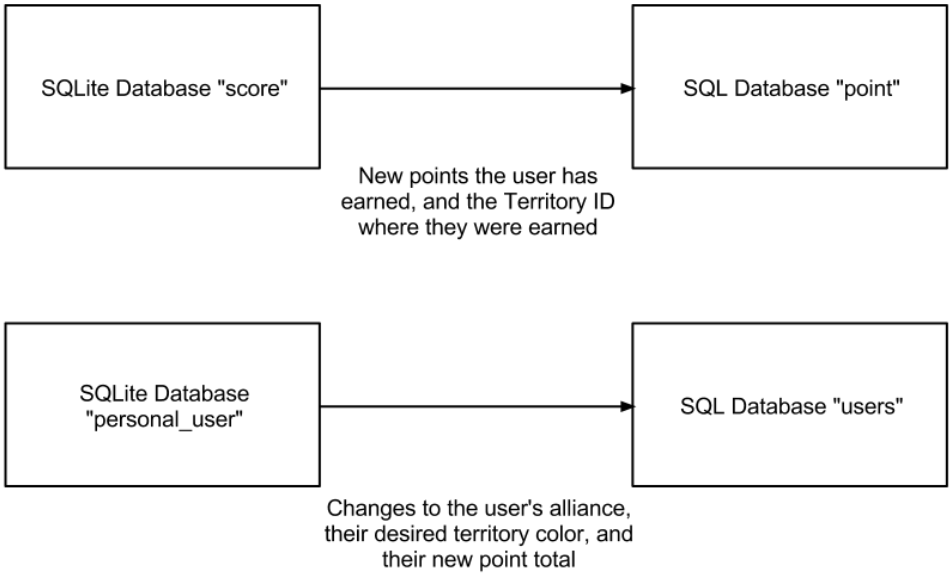


Figure 57: A diagram of the Database Synchronization Upload

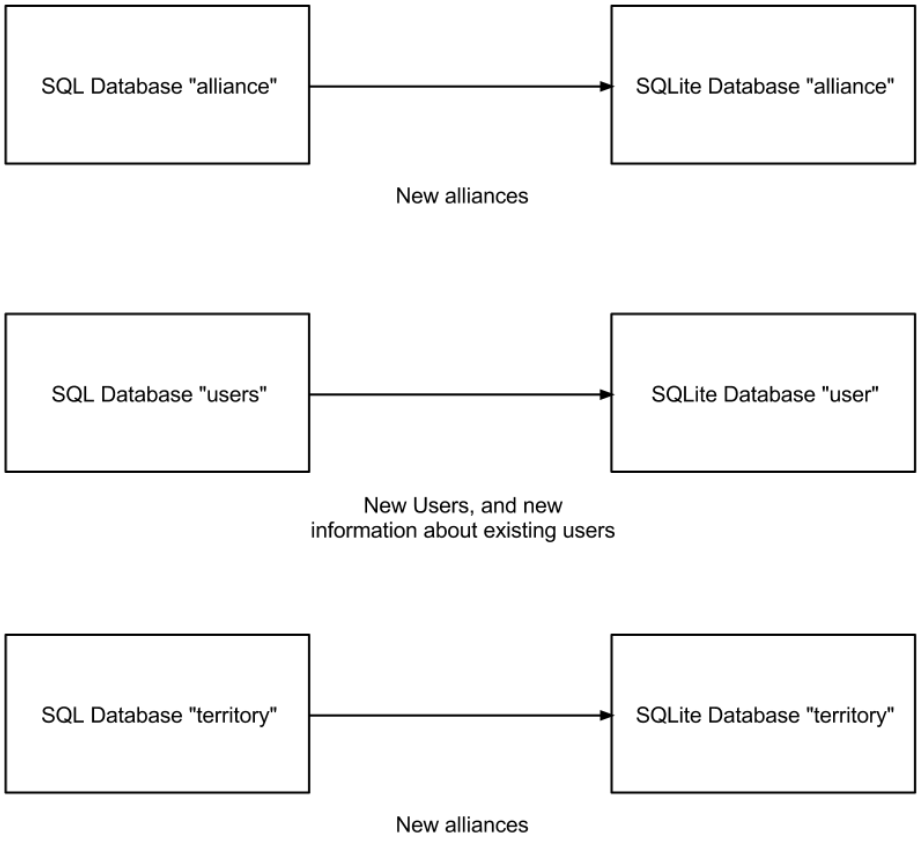


Figure 58: A diagram of the Database Synchronization Pull

As shown in Figure 56, when the user walks with their Android device, the phone's GPS system sends updates to the GPS Manager, a system that keeps track of the current and previous GPS location. When a new location is received, the GPS Manager passes the current and previous GPS location to the Point Calculator. Using the information stored in each GPS location, the Point Calculator determines how many points the user earns from walking between the two points. As part of the calculation, the Point Calculator passes the current GPS location to the Unique Walking Tracker, which determines whether or not the user has been at or near that GPS location before. If they have not, they earn a ten percent bonus on their points for that step. Once the points are calculated, the GPS Manager checks to see if the user is within the same Territory as they were last step, using a method in the Territory class. If they are not, then it checks each of the previous territory's neighbors to see which one the user is in. Once the GPS Manager knows how many points the user has earned, and in which territory those points were earned, the user's point data is updated in the Android's SQLite databases. Once the user wants to upload their scores to the central database, the Sync Manager manages communication between the Android device and the external web server. First, the user's personal information is updated, and then their score data is added, as shown in Figure 57. After that, the Sync Manager also retrieves any data from the central database that has been updated since the last time the user synchronized, and updates the Android's database to reflect those changes, as shown in Figure 58.

These systems will be detailed further in the following sections. Furthermore, any features that were cut will be detailed in Section 7, Conclusions.

5.1 Database Design

In order to store the data generated and used by this application, a MySQL database was created, as well as PHP scripts to manage the data's access and storage. Additionally, the application makes use of the Android's ability to create smaller databases using SQLite.

Table 14: The database schema for the external database "users"

Column Name	Data Type	Description
U_ID	Int, Primary Key	unique ID of the user
Username	varchar(255)	unique username of the user
User_color	int	the int representation of the user's territory color
Alliance_ID	int	the ID of the user's alliance, -1 if they are not part of an alliance
Admin	int	1 if the user is the administrator of their alliance, -1 otherwise
Total_Points	int	The total points this user has earned over all territories
User_Timestamp	Timestamp	The last time this column was edited, MySQL automatically updates this value

To store the users' information, table "users" (see Table 14 above) was created on the external database. A row is created whenever a user makes a new account, and a row is updated whenever that individual user synchronizes their data (see Section 5.2). The SQLite table "user" (see Table 23) pulls its data from this table.

Table 15: The database schema for the external database "alliance"

Column Name	Data Type	Description
Alliance_ID	int, Primary Key	Unique ID for the alliance
Ally_Name	varchar(255)	Unique name of the alliance
Ally_Timestamp	Timestamp	The last time this column was edited, MySQL automatically updates this value

The external database table "alliance" (see Table 15 above) stores the general information for each in-game alliance. Just the alliance's name and unique ID are stored here. Membership information is stored in the "users" table (see Table 14) in the column Alliance_ID. The SQLite table "alliance" (see Table 19) pulls its data from this table.

Table 16: The database schema for the external database table "point"

Column Name	Data Type	Description
P_ID	int, Primary Key	Unique ID for each point entry
U_ID	int	The ID of the user who earned these points
Ter_ID	int	The ID of the territory where these points were earned
Original_Total	int	The original amount of points that were earned

Amount	int	The current amount of points, which degrades over time with respect to territory ownership, does NOT affect the user's point total
Point_Time	Timestamp	The last time these points were altered, this value is automatically updated by updatepoints.php

To handle point degradation for the territories (see Section 5.5), the table "point" (see Table 16) was created on the external database. Whenever a user synchronizes their data (see Section 5.2), one row is created for each territory they have earned points in. This table is used for processing the point data, and it is not directly read from the application. Every week, ten percent of the value in "Original_Total" is subtracted from the value in "Amount". When "Amount" is less than or equal to zero, the row is deleted. When this subtraction takes place, the relevant rows in "Territory_Totals" (see Table 18) are also updated.

Table 17: The database schema for the external database table "territory"

Column Name	Data Type	Description
Ter_ID	int, Primary Key	Unique ID for each territory
Ter_Name	varchar(255)	The territory's name
Ter_Owner	int	The ID of the owner of this territory
Ter_Color	int	The integer representation of the current color of this territory
Ter_Timestamp	Timestamp	The last time this column was edited, MySQL automatically updates this value

Although territories have their areas and neighbors hard-coded on the application (see Section 5.5), their names, owners, and current color are stored on the external database table “territory” (see Table 17). The PHP script updatepoints.php handles assigning ownership of territories – pulling the appropriate color from the table “users” (see Table 14) - and the application pulls this information from the database on data synchronization (see Section 5.2). The application never directly writes to this table.

Table 18: The database schema for the external database table “territory_totals”

Column Name	Data Type	Description
ID	int, Primary Key	Unique ID for this column
U_ID	int	ID of the user with this point total
T_ID	int	ID of the territory where these points were earned
total	int	The total of all the points the given user has earned in the given territory

For ease of programming territory ownership, the table “territory_totals” (see Table 18) was created on the external database. Each row contains the current amount of points a certain user has earned in a certain territory. Note that this is purely for the purposes of territory ownership (see Section 5.5), and has nothing to do with the users’ actual point total. In order to determine who owns a territory with a given T_ID, a SELECT command is run on the table, getting the row with the highest value in the “total” column. The U_ID that is in the returned row is the user ID of the user who now owns that territory.

Table 19: The database schema for the application database table “alliance”

Column Name	Data Type	Description
_id	int, Primary Key	Unique ID for this alliance
ally_name	Text	Unique name for this alliance

The SQLite Database table “alliance” (see Table 19) contains the user’s copy of the general alliance data. This table pulls its data from the external database table “alliance” (see Table 15).

Table 20: The database schema for the application database table “personal_user”. This table keeps track of the logged in users

Column Name	Data Type	Description
_id	int, Primary Key	The unique id of this entry
user_name	Text	The username of this user
user_points	int	This user’s point total
app_start	long	The last time this applicaiton was started (in milliseconds since Jan 1, 1970)
app_end	long	The last time this application was closed (in milliseconds since Jan 1, 1970)

In the application, it became necessary to store more information about the logged-in user than other users. For this purpose, the table “personal_user” was created on the application. A single row is created when the user first logs into the application. As there is currently no way to log out of the application, this table will only ever contain one entry, or no entries. In this way, the application can detect the first time it is turned on.

Table 21: The database schema for the application database table "score"

Column Name	Data Type	Description
_id	int, Primary Key	The unique id for this entry
score_value	int	The points earned in this territory during this walk
score_terr	int	The ID of the territory where these points were earned

The SQLite Database “score” (see Table 21) is used to store the users’ point data. This table is initially empty, and is filled as the user walks. A new row is created each time the user earns points in a territory that is not currently represented in this table. When the user sends their walking data to the external database (see Section 5.2), each row is sent to the table “point” (see Table 16) with the current time as the value for “Point_Time”. Then the table is cleared, so as not to send repeat data on the next synchronization.

Table 22: The database schema for the application database table “territory”

Column Name	Data Type	Description
_id	int, Primary Key	The unique ID for this entry
ter_name	text	The name of this territory
ter_owner	int	The ID of the user who owns this territory, -1 if it is unowned
ter_color	int	The integer representation of this territory’s current color

The SQLite Database “territory” (see Table 22) is used to keep track of the current status of ownership of each territory. This table should be an exact copy of the external database table “territory” (see Table 17), minus the column “Territory_Timestamp”.

Table 23: The database schema for the application database table “user”

Column Name	Data Type	Description
_id	int, Primary Key	The unique ID for this entry
user_name	text	The unique name of this user
user_points	int	The total points earned by this user
alliance_id	int	The id of the alliance this user belongs to, -1 if they do not belong to an alliance

In order to make the leaderboards, the application needs to store some basic information on each user: unique ID, username, total points, and alliance membership. This data is pulled from the external database table “users” (see Table 14). This data is only referenced by the application when creating the leaderboard, and is never directly written to, except when the user first logs in. Then a row is created for the new user, both here and on the external database table “users”.

All of the data created by the programmers and generated by users is stored on the main database hosted on a server external to the application. Each phone with the application keeps a truncated version of the main database. While it may have been possible to store all the data on the users’ phone and a central database, it would have been impractical to do so. Android devices have a relatively limited amount of memory available to store data compared to a server with potentially terabytes of storage space. Additionally, storing more

data with the application means that more data would have to be exchanged with the database during synchronization, which would consume more of the phone's battery power. Therefore the application was designed so that it could perform as needed with a minimal amount of data actually stored on the phone. The data kept on the phone was determined by how much the user needs in order to effectively play the game. For instance, one user does not need to know the status of other users' gradually decreasing points. Instead, the user just need to know who owns a territory and how many total points the owner has.

On Android devices, each individual database needs to have an associated helper class which extends the Android's SQLiteDbHelper class. These classes contain the means to access and alter their respective databases. They accomplish this through the GetReadableDatabase and GetWritableDatabase methods. They also contain helper methods that are automatically called when databases are created. The method most helpful to this application was OnCreate, a function that is automatically called the first time the application gets a readable or writable database from the helper class. This method was used to create the database's structure, and then populate the database with initial data as necessary.

On the server side, it was necessary to create several PHP scripts to let the phone access the database and manage the data. The first script, worldwidewalk.php, allowed communication between the application and the SQL database. This was needed because Android devices cannot directly connect to a SQL database. They require a go-between script that accesses the actual database, and returns the result. The phone first sends a HTTP POST request with various variables the script uses to create the SQL Query.

Table 24: The variables sent to the PHP script along with the various commands

Command	Description	Arguments
SELECT	Selects all the rows from a table	SELECT, the table name
SELECT TIME	Selects only the rows with a more recent Timestamp than is sent	SELECT, the table name, the column name containing the timestamp, a timestamp formatted in the form (‘yyyy-mm-dd hh:mm:ss’)
INSERT	Inserts a row into a table	INSERT, the table name, comma-separated-list of column names, comma-separated-list of column values
UPDATE	Updates a row	UPDATE, the table name, a column name, a column value (For the WHERE clause), comma-separated-list of column names, comma-separated-list of column values
REPLACE	Updates a row, or Inserts one if it is not found (only used on Territory_Totals)	REPLACE, the territory ID, the user ID, the point total

Then the script will reply back with data in JavaScript Object Notation (JSON) form.

Table 25: The data returned from the PHP script with each of the commands

Command	Returns
SELECT	All of the rows from the given table
SELECT TIME	All of the rows from the given table with a Timestamp from after the given Timestamp
INSERT	The row that was inserted
UPDATE	All the rows that were modified
REPLACE	The affected row, or the new row if a new one was created

The second script, updatepoints.php, manages Territory ownership on the server side. This script is scheduled to run at designated intervals on the server. Every four hours, the script runs a SELECT command on the Points database (see Table 16), picking out any entries

that had the script had not modified in the last week. For every row returned, the script removes 10% of the original point total from the current total. It then updates the database with the new point values, in both the Points database and the Territory_Totals (see Table 18) database. Finally, it selects whoever currently owns the most points in the affected territories, and updates the Territory database (see Table 17) with the new owner's information. Moving these calculations to the server side meant fewer load on the phone's limited battery life.

5.2 Database Synchronization

As with any program that stores data on a server, it was important to consider when the application would synchronize its data with the server. In this application, the synchronization process was divided into two parts: the upload and the pull. The first part, the upload, happens when the application sends its data to the database. The only data sent is the walking data the user has generated since the last update.

Table 26: The data that is sent over to the external database

Phone Database	Data Sent
Score	The user's ID ,the territory ID where the points were earned, the Amount, a timestamp of the current time (sent for each row)
Personal_User	The user's current point total, their chosen color, the ID of their alliance Also updates their owned territories with the new color

The phone takes each row of data and creates a DatabaseTask to insert the data into the points table. The phone also takes the user's new point total and updates the corresponding column in

the User table. The final process the phone has to do is update the appropriate tables that handle Territory ownership. This part happens either when the user hits the "Sync" button, or the first time they have a Wi-Fi connection after eight hours of not updating.

The second part happens when the application pulls data from the external database. This part was simple to implement; each individual SQLiteDBHelper class keeps track of when it was last updated. When the time comes to pull from the external database, the application SELECTs all the data that had been updated since that time. The application then UPDATES all of its databases with the rows that had been returned. This happens the first time the application is turned on, and whenever the user uploads their data to the server.

On Android devices, the main UI thread cannot handle network operations; otherwise the application would freeze while waiting for the operation to complete. To solve this problem, an AsynTask was created to handle all database operations called DatabaseTask. On execution, the task takes in strings, depending on the SQL query, and returns arrays of data in JSON format (See section 5.1)

5.3 Map & Routes

The team decided to use map to show where the users are going and simultaneously display walking data. The map also supported the concept of people walking all around the world. There were two options to implement a map on an Android application. One was to use a developer-made map generated for an unusual mechanism in the developer's application. The other was to use Google Maps which was usually an android developers' choice for applications that required a map. Further research showed that GoogleMaps was

sufficiently documented and used by many developers. *World Wide* Walk only needed a map that had an accurate depiction of the world and Google Maps met those needs.

Google Maps was simple to integrate into the application. It uses the MapView class in the Maps external library and provides built-in map downloading, rendering, and caching of Map tiles. This allows developers to work with Maps data similarly with other types of Views in the Android without having to worry about generating the map itself. Since MapView would give access to Google Maps data, developers need to register their application with Google Maps service whether the application was being developed on an Android Virtual Device (AVD) emulator or on an Android device.

Registering for a Google Maps Android API Key came in two parts. The first part was to register the Message Digest algorithm 5 (MD5) fingerprint of the certificate that would be used to sign the application. MD5 is a digital fingerprint which is unique to each application. Google would then provide the Maps API Key associated with that certificate. The second part was to add a reference to the Maps API Key in each MapView declared in XML display view or instantiated directly from the code. *World Wide Walk* declared its Maps API Key in the mainActivity XML. Once these two parts were completed, MapView would download the Map tiles from GoogleMap service and a map would be generated on the Android screen.

There were two Maps API Key Google would generate depending on what MD5 fingerprint was inputted. One was debug mode which allowed Android Virtual Device (AVDs) to generate the map tiles. AVDs are simulated Android devices on the computer and are normally used by developers to debug code if they do not have a physical Android device.

The other was a registered Maps API Key associated with the application and will generate map tiles on a physical Android device unlike the debug Maps API key. The first was used when *World Wide Walk* was starting out. There was not a need to register the application when there was no application to register at the time. As *World Wide Walk* developed and evolved, the Maps API key changed to reflect the need to test-drive the application on a device.

Routes shown on the Map of where the users walked were on a separate Overlay. Overlays are similar to layers on drawing programs like Photoshop or placing tracing paper on top of each other with a single line on each of them and they form a picture. As the Global Positioning System (GPS) updated the user's location (See Section 5.4 Global Positioning System), the new GPS coordinates were added to the overlay and then shown on the screen in form of a red line. The red line was drawn using the Canvas class that drew lines between the arrayList of GPS points. The user's beginning location and current location were indicated by red circles.

5.4 Global Positioning System

To be played by its users, *World Wide Walk* needed to automatically gain information of where the users were currently located. To do this, *World Wide Walk* had to access the GPS functionality in the Android device and certain permissions were required in the AndroidManifest.xml to use the GPS. The AndroidManifest.xml presents essential information about the application to the Android System before any of the application's code could be run. The permissions needed were: *ACCESS_FINE_LOCATION, ACCESS_COARSE_LOCATION,* and *INTERNET.*

Some of the complications that come with using GPS were the accuracy of where the users were. If on a Wi-Fi network, their location provider can be in a different location than they actually are. The other complication was the time it took to get the users' GPS coordinates using the satellites to pinpoint their exact location. This could range from in a couple of seconds to minutes.

World Wide Walk was programmed to use Wi-Fi providers as well, but this created problems when users were inside a building. When indoors, the Wi-Fi provider location would jump, and *World Wide Walk* could not determine how much the users had walked. This was considered cheating as a user could be sitting still and gain points as the Wi-Fi provider location jumps. Thus the program was revised so that users could not gain points while inside a building. Otherwise, the program could detect that the users were outside by using satellites to locate their position.

Some concerns such as accuracy and signal strength tolerances did not crop up. This was because when using the GPS, there are fields in the method in LocationManager (an Android class) called requestLocationUpdates that sets the minimum update distance and minimum update time. The phone regulated the accuracy and signal strength tolerances of GPS as well.

5.5 Territory Design

A major component of this game is the territory system, and such a large, complex system needed careful design. To start with, the area and relative orientation of territories were ultimately created and stored on the application, with their ownership stored on the external database.

The territory class contains an ArrayList of LatLong points, a private helper class. A LatLong point is a structure used to contain a single point in latitude/longitude coordinates for ease of programming. Territories are initially created with only a name and a unique id, with the option of passing in an ArrayList of points to define its area. Alternatively, the points could be manually added one at a time with a method called addVertex, though this runs the risk of the application performing operations on a territory with no defined area. Such a method was only used when the territories were being initially created. A territory's area is defined by drawing a line starting at its first vertex, then continued through the ArrayList of vertices until it drew a line between the final vertex and the initial vertex. Territories also kept an ArrayList of their adjoining neighbors. This allowed minimal searching when locating where the user was. (see section 5.4) Territories also contain the data necessary to display themselves. They kept track of their owner's chosen Territory color, or simply returned a default color if unowned. They also contain a method which returned a Path of their vertices in the correct order, which the application then used to draw the Territory on the map. In this way, users could easily see who owned a particular Territory at a glance. For this version of the application, only two territories were created for testing purposes. More on the future of territories can be found in the "Future Work" section.

Territories also contain a method to determine whether or not a given GPS point was within their boundaries. The Angle Summation method (see Figure 59) is the method used to determine this. For every border line of the territory, the angle between one endpoint, the test GPS point, and the other endpoint is taken.

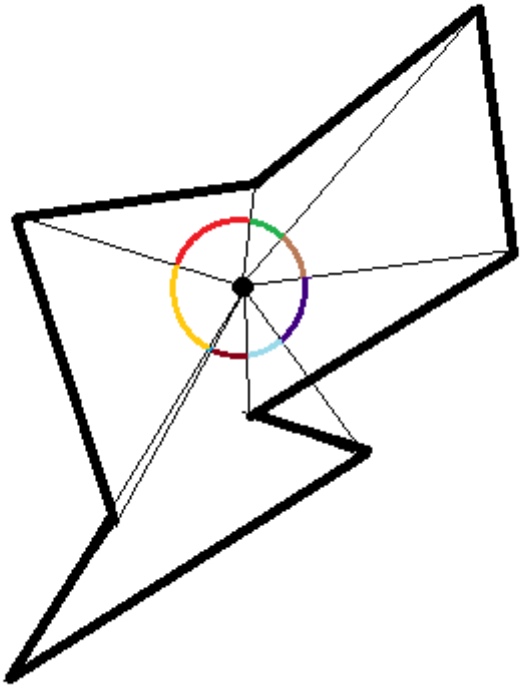


Figure 59: Angle Summation Test. The thick black lines mark the boundary of a territory, and the black dot represents a GPS point. When the angles of the endpoints of all of the sides and the GPS point are summed, the total is 2*pi (as shown by the multicolored circle). Thus, the point is within the territory.

Each of these angles is summed, and if the total is 2*pi, then the point is within the territory. If it is 0, then it was outside the territory (Haines, 1994). While this method is inefficient for shapes over 100 sides, the polygons the application would be working with would have significantly fewer sides (MacMartin 1992). Thus, the negative aspects of this method do not have a large impact on the application. Its conceptual simplicity and reliance upon data that was already being stored made it the clear choice to be implemented for this application.

5.6 Unique Walking Tracker

As part of this game, a system was needed to determine where the user had walked previously. With such a system in place, the game could offer bonus points for walking in new areas in order to encourage users to explore new places and see new sights.

In making such a system, the main consideration was one of accuracy vs. memory usage. This became apparent as the first prototype was developed, which used a series of vectors to determine where the user had walked before. When the GPS returned a new point, the system would look for the closest vector to that point, and determine whether or not the point was within a set distance from the vector, as shown in Figure 60.

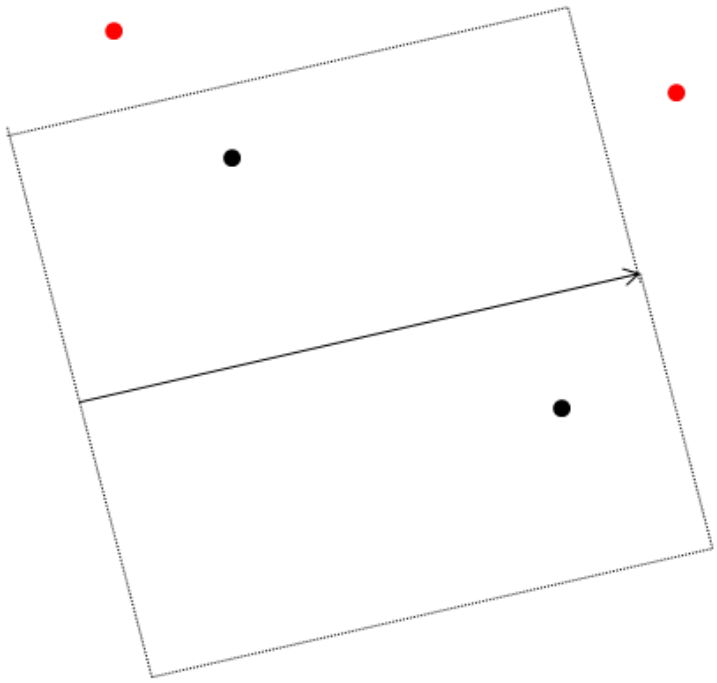


Figure 60: A demonstration of the Vector-based Unique Walking Tracker. Black points are considered within visited areas, red points are within new areas.

If the point is within the region, the system reports that the point was within the area the user had walked before, and nothing more would be done. If not, then a new vector would be created and added to the database using the user's last known location as the starting point, and their current location as the endpoint. While extremely accurate, this system would require vast amounts of processing power to function properly. If the GPS pinged every twenty seconds and the user walked in new territory for an hour, this would create approximately 180 new vectors in a single walk. The user could easily surpass one thousand such vectors in a single week of use. Although methods could have been implemented to make traversing so many vectors simple, and such a number of vectors would not take up a significant amount of memory, those vectors would have to have been stored on the phone's database. Reading and writing to a SQLiteDatabase requires significantly more time and processing power when compared to adding the same data to a data structure like an ArrayList. A query has to be prepared, the data has to be sent to the phone's database, the new row must be created and inserted, and then the database has to return the appropriate value for the query. Having to perform this operation every time a new vector is created could lead to unneeded strain on the processor. Thus, it was determined that a better, more efficient should be sought out.

It became clear that there would have to be a trade-off in accuracy and processor usage, and the next system was designed with this in mind. Additionally, it was realized that the users would, on average, be walking in a small area when compared to the full surface area of the world. With this in mind, the second system was designed around QuadTrees, a method of storing sparse data. Each node in the QuadTree had a width, and stored a Boolean saying whether or not it had been visited before. Nodes also had a minimum width; once a node reached this width, they would stop dividing into smaller nodes and became leaves in the QuadTree. When checking if a new point was in a new location, the tree would simply be

traversed down to a minimum-width node, and the Boolean value of that node would be returned. As this system was conceptually simple, easy to program, and required significantly fewer SQLite insertions, it was selected ideal for the application.

However, this system still had many important design decisions that needed to be made if it was to serve the end-goal. In determining the minimum possible width of a node, it was necessary to consider the time between GPS pings. If the width was low and the time between pings was high, then the user would often walk past entire nodes before the GPS pinged. The users would not receive credit for walking through the areas that they had walked through. The width had to be low enough to ensure accuracy, but high enough to ensure that a user would not regularly skip nodes. In the end, a minimum length of 15 meters was designed to regulate this. Additionally, should the user somehow discover the coordinates of a minimum-width node, then it would be possible to simply walk within that node for hours (as shown in Figure 61), constantly earning bonus points for visiting a new area because the node would not be marked as “visited” until they left.

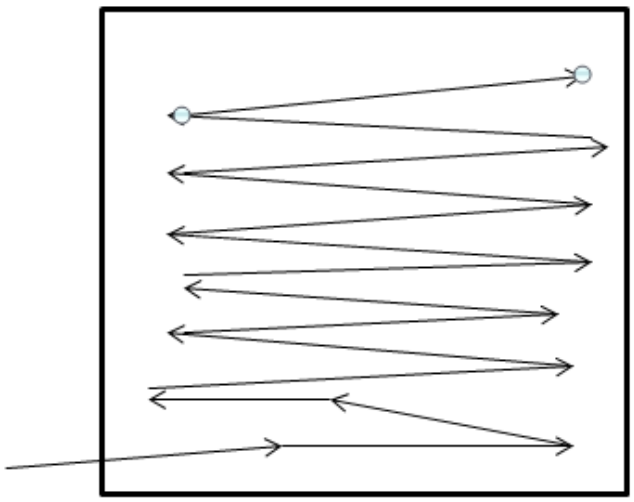


Figure 61: A user uses illicit knowledge to earn bonus points on every arrow. Endpoints determined by GPS location.

The node cannot be marked as “visited” when the user first enters it, as they would not gain bonus points for a few other, legitimate points reached while in the same node. To prevent cheating, nodes keep track of the number of times they have been visited, and after so many visits, they automatically mark themselves as visited. This way the system can maintain a high level of accuracy, while not consuming too much data in the process.

5.7 Point System

The point system was the primary feature to encourage positive habits in the users of this application and thus it was designed around the behaviors the wanted to see in the users. The first such behavior was better, healthier walking. To encourage this, the user earned more points by walking in ways that would burn more calories, such as walking faster or walking uphill. GPS points, called Locations in Android, contained both altitude and timestamp data. Combined with the distance between points, the slope and speed of the user’s walk can be calculated. Walking with greater speeds earns more base points via a stepwise function, as shown by Table 27.

Table 27: Bonus points earned for various walking speeds.

Speed Range (in m/s)	Bonus Points Earned
Speed < 1.3	0
1.3 <= Speed < 1.5	5
1.5 <= Speed < 2.0	10
2.0 <= Speed < 2.5	15
2.5 <= Speed < 3.0	20
3.0 <= Speed < 3.5	30

3.5 <= Speed < 4.0	40
4.0 <= Speed < 5.0	55
5.0 <= Speed < 6.0	70
6.0 <= Speed < 7.0	85
7.0 <= Speed < 8.0	100

Walking up an incline scales that base value by the percentage grade of the slope. In planning this system, it became apparent that users could cheat this system by leaving the application turned on while in a car or other fast-moving vehicle. To prevent this, points are not earned if the phone detects they are moving above 8 m/s. The coach would also reprimand the user if the application detects cheating. In determining this cutoff speed, world-record human land speeds were researched and determined to be 10.4 m/s[12]. This value was then adjusted down to 8 m/s to better fit the average person's maximum speed.

The second behavior was the desire to explore. This was inspired by the adventures the development team experienced while exploring Japan on foot. The user can earn bonus points by walking in real-world locations that they have never walked in before, as determined by the unique walking tracker (see section 5.6). The bonus would be applied as a flat scaling factor after all other point variables had been calculated: 1.1 if they are walking in a new location, and 1.0 otherwise. This gave users an incentive to walk different routes, even if they walk to the same destinations every day.

The third behavior to encourage was social interaction via walking. This was mainly accomplished through the territory and alliance system. Due to the need for constant walking to

[12] *100 M Men's Final Results*. N.p.: IAAF, 16 Aug. 2009. Pdf.

maintain ownership of a territory (as outlined in section 5.5), owning a territory is hard work, especially with many others competing for the same area. Owning more than one territory is even harder. By joining an alliance, a users' potentially small area can become part of a large communal area. If everybody in an alliance owned just a single territory, then the alliance could potentially own quite a large area, and each user would feel as though they were contributing to this success. This behavior was also encouraged through the leaderboard system. As everybody could see if a user dropped in position, there were social pressures to perform well. Additionally, a user's points could contribute to their alliance's total points, which would make the user feel partially responsible for their alliance's leaderboard.

5.8 Coach Manager

One of the most complicated coding was the Coach Manager. The Coach Manager was integrated throughout the program in many locations. It regulated when the coaches were to appear and made sure the correct coach appeared with the correct text by pulling the coach text information from a pre-programmed database. Table 28 shows the database that contains the data for each coach.

Table 28: The database schema for the application database table "coaches"

Column Name	Data Type	Description
_id	Int, Primary key	Unique ID
coach_name	Text	The name of the coach
Style	Text	The style of the coach
Intro	Text	Introduction text
App_start	Text	Application start text

No_walk	Text	Text for when the user has not walked for a certain amount of time
Walking	Text	Text for when the users has walked non-stop for a certain amount of time
Cheat	Text	Text for when users cheat by “walking” too fast
Complete_80	Text	Text for when users are 80% towards completing a mission
Complete	Text	Text for when users have completed a mission
Territory_cap	Text	Text for when users captured a territory
Territory_lost	Text	Text for when users lose a territory
Territory_ally_cap	Text	Text for when the alliance user is in captured a territory
Territory_ally_lost	Text	Text for when the alliance user is in lose a captured territory
New_member	Text	Text for when new members join the users’ alliance. This would be appended with the number of users that have joined the alliance.
Member_lost	Text	Text for when members of the users’ alliance leave.
No_walk_number	Integer	Contains the number in days where coach yells at users for not walking
Non_stop_walking_number	Integer	Contains the number in minutes where coach praise users for non-stop walking

The correct images that needed to appear were preprogrammed in the code. Most conditions had one or two images so a simple random integer of 0 or 1 determined which image would show on the screen.

A full table of art assets is listed in Appendix A.

The main problem with creating Coach Manager stemmed from Android inability to share Views. If a view has already been created by an activity it cannot be touched by another activity or class. This included background tasks. Without having access of Views, coaches could not appear on screen when needed. Thus some hardcoding in the mainActivity had to be implemented for Coach Manager to work.

When conditions were met the coaches were supposed to appear with the appropriate text. Some of these conditions are event-based and some are time-based. For the event-based items, the app has a method called coachPopup which is placed in the code where it is needed, for instance after a user is created and the user chooses a coach or when the user starts the application again. CoachPopup calls for the CoachManager to pull information from the database and input the information in the correct view of the XML layout.

When the application started up again, the app checked the difference in days of when the user closed the app to when the user started the application previously. Both of these datas are pulled using the UserManager. If the user had not walked a certain amount of days, determined by the coach's personality, then the noWalking method is called which then calls coachPopup.

The event-based items are situated in a TimerTask which runs periodically. After a minute has passed, the method checkStatus checks if certain conditions are met. If the user has walked non-stop for a certain amount of hours or minutes, different to each coach, coachPopup is called. CheckStatus also checks if the user is moving at a speed that is too fast and will trigger the coach to popup with the appropriate response. It also regulates the time when missions

popup for users to accept or reject. These missions are given after a random() generated integer from 1 to 100 amount of minutes has passed and if the users does not have a mission. It also checks for when missions are 80% complete or fully completed which also triggers coachPopup. If none of the conditions are met checkStatus lies dormant until it is called again.

Some conditions had to be programmed in a more complex method. For instance, finding out if the user has gained or lost a territory required access to the sync method, Users Manager, and Territories. The code needed to be integrated into the Sync button so that it could gain the necessary information of the Territories data that is transferred between the central database and the phone's database. Through the Territories data, Coach Manager then searches for the user's unique ID stored in the Territory database and pulls the information into an array list. Then the list is compared with a previous list, checking for different conditions and the appropriate response is programmed for the coach to appear.

Some of the coach manager structure is hardcoded in the application due to complications with sharing Views between activities and classes. It is one of the few pieces that interact with many databases and classes. Some of the databases that it uses are personalUserDatabase (the database on the phone), TerritoryDatabase (when synced with the central database), and UserDatabase (the database that contains information about all users).

6 Testing

The application was downloaded to a tablet running version 3.2 of Android, and tested in the neighborhood of Eirakusou, where the team stayed in Japan. The indoor environment was a regular 2-floor Japanese house and the outdoor area contained both slopes and flat roads. The team ran through all the functions of the application and found the following results shown in Table 29.

Table 29: Testing and Result

Test	**Method**	**Result**
UI testing	Click on buttons to test UI (tap to start, call out and dismiss coach)	UI functions as intended
Create new user	Type in new user name in the user creation window and hit confirm	New user can be created, and is entered into the database as expected
Create an alliance and join an alliance (multiple users were created to test this feature)	Type in new alliance name in the user creation window and hit confirm	New alliance is created Simulated users can join the newly created alliance
Indoor walking	Walk inside the Eirakusou house and within the laboratories of Osaka University	GPS has difficulty getting a signal, and often does not function at all
Outdoor walking	Walk outside	As the user walks, their path is drawn on the map and points are collected
Coach switch	Select a new coach from the coach selection screen and hit the “confirm” button	The new Coach appears on the screen and the UI style is changed to match the coach’s art style
Sync	Click on the “Sync” button on	Syncs phone database with

	the main toolbar.	central database. This is shown by changes in the leaderboard and territory ownership, as well as observing the data within the central database.
Switching device orientation	Turn the device horizontal and vertical	Initially restarts the application. Now it is fixed so that the essential game features are persistent.
Testing coach pop-ups under the following conditions		
Application start	Turn on the application after closing it.	Coaches use their daily greeting
First meeting	Start a fresh game, then select a new coach	Coaches greet users with a self-introduction.
No walking in “X” days	Stay inactive for “X” days	Coaches pop up at startup and remind the user
Non-stop walking for “Y” hours/mins	Keep walking for “Y” hours/mins	Coaches pop up and encourage the user
Walking speed too fast (in a car, biking, running, etc.)	Bike/drive with the device	Coaches pop up and notify the user
Mission 80% complete	Start a mission and keep walking until the mission is 80% complete (the mission was to collect 100 points and the team walked until 80 points were collected)	Coaches pop up and encourage the user
Mission complete	Complete the mission	Coaches pop up and encourage the user
Territory captured	Gather the most points among other simulated users in the Osaka University Territory	Coaches pop up and congratulate the user
New member joined alliance	Created a member directly into the database that is joined to the alliance.	Coaches notify the user and the number of users that have joined user’s alliance

Member left alliance	Directly manipulated the database so that another user is not in an alliance or moved to another alliance.	Coaches notify the user and the number of users that have left user's alliance

All the interface and GPS functions were working as expected. The team tested ten coach responses for each coach as well as switching coach mid-way during a walk. For instance, if a user switched their coach to a different one, the coach pop up would follow the setting of the new coach. For missions, changes in coaches would not affect when users get a mission, nor the progress of an ongoing mission.

While the testing showed that the application successfully executed most of its intended functions, there are other features that were unable to be thoroughly tested due to lack of time.

The following features were not tested:

Territory lost, Territory captured/lost (alliance), Point notification

7 Conclusion

This section discusses the problems that the team faced during the project and summarizes the final outcome.

7.1 Difficulties

The team has achieved most of its goals. It made a game that runs smoothly on Android phones, although the effectiveness of the users' motivations was never thoroughly tested. Some difficulties were presented during development, including unexpected program crashes, bugs, a lack of online support for the Google Maps API, and a few redesigns of some of the systems. In retrospect, the team could have adjusted their schedule to allow for more time to recover from these delays. Although not as complicated as the technical side, the art team had to overcome their own set of obstacles. The artists' main challenge was to come up with sophisticated, original character designs. By the end of the project, the team had created twelve distinctive characters/coaches, each with eight different hand-drawn expressions. The team was proud of the end result when all the assets were put together and the team believed that the coaches added personality to the game.

The team also encountered another significant difficulty in the development of *World Wide Walk.* A major part of the application involved using the Google Maps API to draw the user's real-world position (see the Maps section). Unfortunately, two months before development started, Google updated their APIs. As a result, a majority of the online help and documentation was useless to us, and the process of developing the map features of this program took longer than anticipated. In the end, the team had to cut out all of the routing features planned to be included in the application. The ability to display territories on the map was also not fully implemented until the

final week of development, when it had been slated as an early feature. As a result, the team was unable to give users the ability to upload their own textures for their territories, and the team was unable to allow them to switch off the territory overlay. Both had been planned features from the start of the application.

In spite of these difficulties, the team was able to develop a functioning application. With more time, more user testing could have been completed, and a greater understanding of the users' motivations could have been reached. But as an initial prototype, *World Wide Walk* was a success. It proved that these technologies could be brought together to develop an exercise game unlike any that has been seen on smartphones yet.

7.2 What Worked & What Didn't

The development team did a good job of testing the individual systems as they were developed, rather than waiting for the application to be pieced together. As a result, many of the bugs were fixed by the time the team reached the final testing phase. Unfortunately, the strategy of piecemeal development meant that the application existed as a series of disconnected features for a majority of the project time. This made it difficult to accurately gauge the team's progress. Once the application came together, development sped up tremendously. Were this project to be done again, the team would work on having a functioning application as soon as possible in addition to thoroughly testing each feature as they were developed.

7.3 Summary

The *World Wide Walk* team has successfully developed a walking exercise game without many of the flaws apparent in other pedometer games. By tracking the users' position directly using a smartphone's GPS instead of using a step-tracking algorithm, the game becomes nearly impossible to cheat. In addition, using the GPS allows the reward system to be directly tied into where the user is walking. Not only does this make the reward system hard to circumvent, it gives the game a solid connection between the virtual and real world. This connection, when combined with such game features as Territory Ownership, Leaderboards, Alliances, and the Coach System, creates many different motivations for users to walk. As existing pedometer applications have shown, gaming elements provide effective motivation for users to use them. The game also provides a wide range of visual choices through its UI and NPC coaches to appeal to a wide range of users. With additional optimization and testing, *World Wide Walk* could become a game that truly makes exercise fun and engaging for users across the globe.

Appendix A: Artistic Assets

All assets are digitally painted.

Table 30: Coach Assets List (.png files)

Name	Description	Name	Description
Draco1	Normal	Draco2	Happy
Draco3	Sad	Draco4	Angry
Draco5	Normal2	Draco6	Angry2
Draco7	Sad2	Draco8	Happy2
Tom1	Normal	Tom2	Happy
Tom3	Sad	Tom4	Angry
Tom5	Normal2	Tom6	Angry2
Tom7	Sad2	Tom8	Happy2
Steve1	Normal	Steve2	Happy
Steve3	Sad	Steve4	Angry
Steve5	Normal2	Steve6	Angry2
Steve7	Sad2	Steve8	Happy2
Samantha1	Normal	Samantha2	Happy
Samantha3	Sad	Samantha4	Angry
Samantha5	Normal2	Samantha6	Angry2
Samantha7	Sad2	Samantha8	Happy2

Sabrina1	Normal	Sabrina2	Happy
Sabrina3	Sad	Sabrina4	Angry
Sabrina5	Normal2	Sabrina6	Angry2
Sabrina7	Sad2	Sabrina7	Happy2
Rick1	Normal	Rick2	Happy
Rick3	Sad	Rick4	Angry
Rick5	Normal2	Rick6	Angry2
Rick7	Sad2	Rick8	Happy2
Aldin1	Normal	Aldin2	Thinking
Aldin3	Laughing	Aldin4	Laughing2
Aldin5	Shouting	Aldin6	Thinking2
Aldin7	Shouting2	Aldin8	Normal2
Catherine1	Normal	Catherine2	Angry
Catherine3	Instructing	Catherine4	Smiling
Catherine5	Clapping	Catherine6	Speaking
Catherine7	Speaking2	Catherine8	Speaking3
Jack1	Normal	Jack2	Burning
Jack3	Scratching	Jack4	Scratching2
Jack5	Happy	Jack6	Smiling
Jack7	Speaking	Jack8	Yeah-posing

Rebecca1	Normal	Rebecca2	Speaking
Rebecca3	Speacial	Rebecca4	Serious
Rebecca5	Instructing	Rebecca6	Sighing
Rebecca7	Smiling	Rebecca8	Smiling2
Rob1	Normal	Rob2	Cool
Rob3	Cool2	Rob4	Serious
Rob5	Shouting	Rob6	Shouting2
Rob7	Thumbs-up	Rob8	Thinking
Toddo1	Normal	Toddo2	Angry
Toddo3	Down	Toddo4	Opening refrigerator
Toddo5	Shrugging	Toddo6	Sad
Toddo7	Smiling	Toddo8	Instructing

Table 31: UI Assets List (.png files)

Name	Function
Coach_Eastern	Switch coach on/off
Coach_Western	Switch coach on/off
Logo	Startup logo
OK_Eastern	Confirm
OK_Western	Confirm

Cancel_Eastern	Cancel
Cancel_Western	Cancel

Appendix B: Project Flyer

Figure 62: Flyer for World Wide Walk at the beginning of the project

Reference/Bibliography

29soft.com, “Diner Dash: Hometown Hero”, 29soft.com 20 Jun. 2012 <http://www.29soft.com/download/diner-dash-hometown-hero/>.

GO-Gulf.com, “SMARTPHONE USERS AROUND THE WORLD - STATISTICS AND FACTS [INFOGRAPHIC]”. *GO-Gulf.com*. 2 Jan. 2012 <http://www.go-gulf.com/blog/smartphone>

Haines, Eric, "Point in Polygon Strategies," *Graphics Gems IV*, ed. Paul Heckbert, Academic Press, p. 24-46, 1994.

itunes.apple.com, “Free Pedometer by Arawella Corporation*”, itunes.apple.com*. 1 Jun. 2012 <https://itunes.apple.com/app/pedometer-free/id362306160?mt=8>

Julie. “Striiv Smart Pedometer Review”. *The Gadgeteer.* 7 May, 2012 <http://the-gadgeteer.com/2012/05/07/striiv-smart-pedometer-review/>

MacMartin, Stuart, et al, "Fastest Point in Polygon Test," *Ray Tracing News 5(3*), 1992 <http://tog.acm.org/resources/RTNews/html/rtnv5n3.html#art3>.

play.google.com, “Accupedo Pedometer”, *play.google.com*, 20. Jul. 2012 <https://play.google.com/store/apps/details?id=com.corusen.accupedo.te>

Salman, Latif, “Angry Birds set to land on Xbox, PS3 and Nintendo 3DS on September 28th”, www.ubergizmo.com, 25 Sept. 2012 <http://www.ubergizmo.com/2012/09/angry-birds-set-to-land-on-xbox-ps3-and-nintendo-3ds-on-september-28th/>

Scarlett, Jim. "Enhancing the Performance of Pedometers Using a Single Accelerometer."*www.analog.com*. Analog Devices, n.d. Web. 17 Oct. 2012. <http://www.analog.com/static/imported-files/application_notes/47076299220991AN_900.pdf>.

Schiesel, Seth. "VIDEO GAME REVIEW | POKéWALKER; Look, Kids: A Way to Slip Pokémon Past Mom." *The New York Times*. The New York Times, 19 Mar. 2010. Web. 17 Oct. 2012. <http://www.nytimes.com/2010/03/19/arts/television/19pokemon.html?_r=1>.

Stuart, Keith. "The Pokemon Pedometer: Childhood Obesity Is so over." *The Guardian*. Guardian News and Media, 21 May 0012. Web. 17 Oct. 2012. <http://www.guardian.co.uk/technology/gamesblog/2010/jan/12/games-nintendo>.

100 M Men's Final Results. N.p.: IAAF, 16 Aug. 2009. Pdf. <http://berlin.iaaf.org/documents/pdf/3658/AT-100-M-f--1--.RS1.pdf>